In compliance with the
Canadian Privacy Legislation
some supporting forms
may have been removed from
this dissertation.

While these forms may be included
in the document page count,
their removal does not represent
any loss of content from the dissertation.

University of Alberta

Fatigue Analysis and Life Extending Control of a Boiler-Turbine System

by

Donglin Li

A thesis submitted to the Faculty of Graduate Studies and Research in partial fulfillment of the requirements for the degree of **Master of Science**.

Department of Electrical and Computer Engineering

Edmonton, Alberta
Fall 2003

National Library of Canada
Acquisitions and Bibliographic Services
395 Wellington Street
Ottawa ON K1A 0N4
Canada

Bibliothèque nationale du Canada
Acquisisitons et services bibliographiques
395, rue Wellington
Ottawa ON K1A 0N4
Canada

Your file Votre référence
ISBN: 0-612-87809-0
Our file Notre référence
ISBN: 0-612-87809-0

The author has granted a non-exclusive licence allowing the National Library of Canada to reproduce, loan, distribute or sell copies of this thesis in microform, paper or electronic formats.

L'auteur a accordé une licence non exclusive permettant à la Bibliothèque nationale du Canada de reproduire, prêter, distribuer ou vendre des copies de cette thèse sous la forme de microfiche/film, de reproduction sur papier ou sur format électronique.

The author retains ownership of the copyright in this thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without the author's permission.

L'auteur conserve la propriété du droit d'auteur qui protège cette thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou aturement reproduits sans son autorisation.

Canada

University of Alberta

Library Release Form

Name of Author: Donglin Li

Title of Thesis: Fatigue Analysis and Life Extending Control of a Boiler-Turbine System

Degree: Master of Science

Year this Degree Granted: 2003

Permission is hereby granted to the University of Alberta Library to reproduce single copies of this thesis and to lend or sell such copies for private, scholarly or scientific research purposes only.

The author reserves all other publication and other rights in association with the copyright in the thesis, and except as herein before provided, neither the thesis nor any substantial portion thereof may be printed or otherwise reproduced in any material form whatever without the author's prior written permission.

Date: Sept. 29, 2003

University of Alberta

Faculty of Graduate Studies and Research

The undersigned certify that they have read, and recommend to the Faculty of Graduate Studies and Research for acceptance, a thesis entitled **Fatigue Analysis and Life Extending Control of a Boiler-Turbine System** submitted by Donglin Li in partial fulfillment of the requirements for the degree of **Master of Science**.

Dr. Tongwen Chen
Co-Supervisor

Dr. Horacio J. Marquez
Co-Supervisor

Dr. Qing Zhao

Dr. Ming J. Zuo
External Examiner

Date: Sept. 29. 2003

Abstract

Traditional control system design focuses on stability and performance only; in other words, the control design process ignores the effects of aging, fatigue, and damage in the materials involved. Motivated by this, in the 1990's life extending control (LEC) appeared. Its main purpose is to extend the life of critical plant components, or to reduce maintenance cost, while simultaneously ensuring that a set of specific performance requirements are satisfied.

In this research, two main topics in LEC, damage modeling and LEC design, are studied for a boiler-turbine system. The following is a list of the research contributions:

- Two new continuum fatigue modeling theories are discussed, and an improved Rainflow cycle-counting method is applied for on-line model-based life extending control.
- A new LEC control system structure is proposed.
- System identification method is used to reduce system complexity.
- Linear quadratic regulator and model predictive control are used to design LEC for a boiler system.
- Simulations are presented with linear and nonlinear models.

To my family

Acknowledgements

I would like to thank my supervisors, Drs. Tongwen Chen and Horacio J. Marquez for their invaluable encouragement, support and guidance on all aspects of this thesis. It has been a privilege for me to work with them. Without their guidance, this work would not have been possible.

I would also like to express my gratitude to the COURSE Program with the Alberta Energy Research Institute for financing this research.

I gratefully acknowledge the support provided by Syncrude Canada personnel; Mr. R. Kent Gooden and Mr. M. Riaz. Kent Gooden initially brought the idea of "life extending control" that eventually lead to this thesis, and closely followed the progress of this research.

I would like to thank the rest of my examining committee, Dr. Qing Zhao and Dr. Ming J. Zuo, for taking the time to read my thesis and for their comments and suggestions.

I would also like to thank my fellow graduate students and friends, for their assistance and support.

Finally, I thank my family for all their support and encouragement through the years.

Contents

1 Introduction 1
- 1.1 Fatigue analysis . . . 1
- 1.2 Life monitor and life predictor . . . 2
- 1.3 The development of life extending control . . . 2
- 1.4 Our research . . . 8

2 Fatigue Damage Modeling 11
- 2.1 The basis of fatigue damage modeling: Cycle-dependent modeling . . . 11
- 2.2 Ray's modeling method . . . 13
  - 2.2.1 Ray's method . . . 13
  - 2.2.2 Rainflow cycle-counting . . . 15
  - 2.2.3 Improved Rainflow cycle-counting . . . 20
- 2.3 Lorenzo's modeling method . . . 24
  - 2.3.1 Zero mean stress continuum damage model . . . 25
  - 2.3.2 Non-zero mean stress Continuum damage model . . . 28
- 2.4 Life prediction . . . 30
- 2.5 Comparison of two methods . . . 30

3 The Controlled Plant 32
- 3.1 Brief introduction to the controlled plant . . . 32
  - 3.1.1 Header system in Syncrude . . . 32
  - 3.1.2 Utility Boiler . . . 33
- 3.2 SYNSIM model . . . 34
- 3.3 Linearized model . . . 35

4 Life Extending Controller Design with Linear Quadratic Regulation Theory 37
- 4.1 Problem formulation . . . 37
- 4.2 Design of an observer . . . 39
- 4.3 Linear quadratic optimal algorithm . . . 41
- 4.4 Simulation results . . . 43

5 Life Extending Controller Design with Model Predictive Control Theory 48
- 5.1 The setup of our control system . . . 48
- 5.2 The reduction of the model . . . 49
  - 5.2.1 The identification of g_{11}, g_{21} and g_{31} . . . 52
  - 5.2.2 The identification of g_{12}, g_{22} and g_{32} . . . 59
  - 5.2.3 The identification of g_{13}, g_{23} and g_{33} . . . 66

5.3 MPC design and simulation 74

6 Conclusions and Future work **81**

Bibliography **83**

List of Figures

1.1 A schematic diagram of LEC 3
1.2 Another schematic diagram of LEC 4
1.3 Combining LEC and existing controller 8
1.4 A new LEC closed-loop scheme setup 9

2.1 S-N curve for some material 11
2.2 Demonstration of nonlinear fatigue damage modeling 12
2.3 Elastic and plastic strains 14
2.4 Variable amplitude strain history 17
2.5 Rearrangement of the strain history 17
2.6 Step 1 of Rainflow cycle-counting 17
2.7 Step 2 of Rainflow cycle-counting 17
2.8 Step 3 of Rainflow cycle-counting 18
2.9 Step 4 of Rainflow cycle-counting 18
2.10 Step 5 of Rainflow cycle-counting 18
2.11 Step 6 of Rainflow cycle-counting 18
2.12 Step 7 of Rainflow cycle-counting 19
2.13 Step 8 of Rainflow cycle-counting 19
2.14 Step 9 of Rainflow cycle-counting 19
2.15 Step 10 of Rainflow cycle-counting 19
2.16 Step 11 of Rainflow cycle-counting 20
2.17 Step 12 of Rainflow cycle-counting 20
2.18 Step 13 of Rainflow cycle-counting 21
2.19 Step 1 of the improved Rainflow cycle-counting 21
2.20 Step 2 of the improved Rainflow cycle-counting 22
2.21 Step 3 of the improved Rainflow cycle-counting 22
2.22 Step 4 of the improved Rainflow cycle-counting 22
2.23 Step 5 of the improved Rainflow cycle-counting 22
2.24 Step 6 of the improved Rainflow cycle-counting 22
2.25 Step 7 of the improved Rainflow cycle-counting 22
2.26 Step 8 of the improved Rainflow cycle-counting 23
2.27 Step 9 of the improved Rainflow cycle-counting 23
2.28 Step 10 of the improved Rainflow cycle-counting 23
2.29 Step 11 of the improved Rainflow cycle-counting 23
2.30 Step 12 of the improved Rainflow cycle-counting 23
2.31 Step 13 of the improved Rainflow cycle-counting 23
2.32 Step 14 of the improved Rainflow cycle-counting 24
2.33 Step 15 of the improved Rainflow cycle-counting 24

2.34 Strain-stress curve under zero mean stress 27
2.35 Strain-stress curve under non-zero mean stress 29

3.1 A schematic diagram of the steam header system with Syncrude 33
3.2 The operating principle diagram of the utility boiler 34

4.1 Comparison of states of the system and observer (states 1 to 5) 40
4.2 Comparison of states of the system and observer (states 6 to 10) 40
4.3 Comparison of states of the system and observer (states 11 to 15) 41
4.4 Comparison of states of the system and observer (states 16 to 19) 42
4.5 Step response of the drum level with the linear model 44
4.6 Step response of the drum pressure with the linear model 44
4.7 Step response of the steam temperature with the linear model 45
4.8 Comparison of the accumulated damage with the linear model 45
4.9 Step response of the drum level with the nonlinear model 46
4.10 Step response of the drum pressure with the nonlinear model 46
4.11 Step response of the steam temperature with the nonlinear model 47
4.12 Comparison of the accumulated damage with the nonlinear model 47

5.1 Input u_1 52
5.2 Input u_2 52
5.3 Input u_3 52
5.4 Output y_1: Drum level 52
5.5 Output y_2: Drum pressure 53
5.6 Output y_3: Steam temperature 53
5.7 The first reference input and drum level output 54
5.8 After segmentation and detrend of the input and output 54
5.9 The first reference input and drum pressure output 54
5.10 After segmentation and detrend of the input and output 54
5.11 The first reference input and steam temperature 55
5.12 After segmentation and detrend of the input and output 55
5.13 The output fit of g_{11} 55
5.14 The residual test for g_{11} 56
5.15 The output fit of g_{21} 56
5.16 The residual test for g_{21} 57
5.17 The output fit of g_{31} 57
5.18 The residual test of g_{31} 58
5.19 Input u_1 59
5.20 Input u_2 59
5.21 Input u_3 60
5.22 Output y_1: Drum level 60
5.23 Output y_2: Drum pressure 60
5.24 Output y_3: Steam temperature 60
5.25 The second reference input and drum level output 61
5.26 After segmentation and detrend of the input and output 61
5.27 The second reference input and drum pressure output 62
5.28 After segmentation and detrend of the input and output 62
5.29 The second reference input and steam temperature 62

5.30 After segmentation and detrend of the input and output 62
5.31 The output fit of g_{12} 63
5.32 The residual test for g_{12} 64
5.33 The output fit of g_{22} 64
5.34 The residual test for g_{22} 65
5.35 The output fit of g_{32} 65
5.36 The residual test for g_{32} 66
5.37 Input u_1 67
5.38 Input u_2 67
5.39 Input u_3 67
5.40 Output y_1: Drum level 67
5.41 Output y_2: Drum pressure 67
5.42 Output y_3: Steam temperature 67
5.43 The third reference input and drum level output 68
5.44 After segmentation and detrend of the input and output 68
5.45 The third reference input and drum pressure output 69
5.46 After segmentation and detrend of the input and output 69
5.47 The third reference input and steam temperature 69
5.48 After segmentation and detrend of input and output 69
5.49 The output fit of g_{13} 70
5.50 The residual test for g_{13} 71
5.51 The output fit of g_{23} 71
5.52 The residual test for g_{23} 72
5.53 The output fit of g_{33} 72
5.54 The residual test for g_{33} 73
5.55 Step response of the drum level with the linear model 75
5.56 Step response of the drum pressure with the linear model 76
5.57 Step response of the steam temperature with the linear model 77
5.58 Comparison of the accumulated damage with the linear model 78
5.59 Step response of the drum level with the nonlinear model 78
5.60 Step response of the drum pressure with the nonlinear model 79
5.61 Step response of the steam temperature with the nonlinear model 79
5.62 Comparison of the accumulated damage with the nonlinear model 80

List of Tables

2.1 Nomenclature of some material constants 31

5.1 Fit rate and order reduction situation 74

Chapter 1

Introduction

1.1 Fatigue analysis

Metal fatigue is a process that causes premature failure or damage of a component subjected to repeated loading [3]. During the first half of the 19th century, the development of the steam engine led to increasing sources of repeated stress on metal parts and structural elements. Shortly thereafter, "unexplainable" fractures—particularly in locomotive axles—became a great concern to engineers. About the middle of the century, experiments of Woehler and others showed the importance of the number of repetitions of stress, rather than duration of time, in causing failure of metals under relatively low stress [19]. Since then over 20,000 papers on fatigue have been published throughout the world [36]. Most of the research work considers the progress of fatigue as occurring in three stages:

1. Start of a crack;
2. Propagation of the crack;
3. Final rupture of the piece.

Most attention has been devoted to the first stage. There are three important theoretical approaches which are illustration of numerous attempts that have been made to explain the development of cracking under repeated stressing. Orowan [33] proposed a simplified model comprising a "plastic inhomogeneity" within elastic surroundings; Freudenthal [16] discussed qualitatively a possible mechanism of crack formation by fragmentation of crystallites; Machlin [27] suggested a theory based on an even more fundamental approach. A large amount of experiments have been done to provide useful data under various conditions. These data constitute the basis of fatigue analysis and system design. But the main objective of early studies was to understand material properties in order to design reliable systems.

1.2 Life monitor and life predictor

Since the 1970's, fatigue research has also been used as the basis for the development of life-prediction systems. Many methods for estimating fatigue life were proposed such that life-monitor systems can be developed [11]. For example, Socie [45] proposed using local stress-strain concepts to predict fatigue life, Dowling [12] made the research of fatigue life prediction under complex load and special metal shape. Based on these theories, Lu [26] developed an on-line stress calculation and life monitoring system for boiler components; Schucktanz [43] an on-line fatigue monitoring system of reactor components; Wilson [53] stress monitoring systems for safeguarding boiler components during rapid temperature transients. However we should notice that life monitor or prediction is just to show the situation of fatigue, so there is no significant difference between them and fatigue analysis in view of the control.

1.3 The development of life extending control

Until the 1990's, control application reported in the literature were focused on stability and performance only, assuming that the materials involved had invariant properties; in other words, the control design process ignores the effects of aging, fatigue, and damage in materials. However, the material involved inevitably suffers fatigue and even damage, and fatigue and damage of some critical components may have negative impact on the performance of the system, shorten the maintenance cycle even the service life of the system, and may lead to serious accidents. Therefore systems designed in this way may lose performance or even stability in the long run, yielding high risks in system operations.

In 1991, Noll and co-workers [32] pointed out the need for addressing the trade-off between system performance and durability (of critical components). This formed the basis for the so-called life extending control (LEC). LEC is an area of research involving the integration of two distinct disciplines: system sciences and mechanics of materials. Although a significant amount of research has been conducted in each of the individual areas of control & diagnostics and analysis & prediction of materials damage, integration of these two disciplines has not received much attention [37]. At present, there is little literature available which directly discusses the combination of these two disciplines. In 1994, Ray and co-workers [38] first introduced some important concepts of life extending control.

Fig. 1.1 is a structure schematic digram proposed by Ray, it shows a conceptual view of

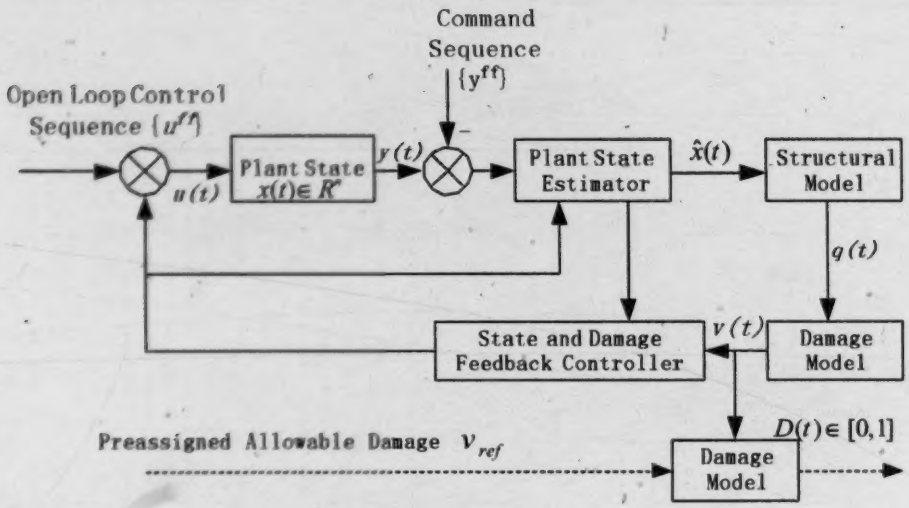

Figure 1.1: A schematic diagram of LEC

the life extending control system, where

x is the plant state vector;

$\hat{x}$ is the estimated plant state vector;

y is the plant output vector;

y^{ff} is the reference plant output vector;

u is the control input vector;

u^{ff} is the reference control input vector;

q is the load vector;

v is the damage state vector.

Because it cannot be guaranteed that all states are measurable, the state estimator is needed. The damage dynamics $v(t)$ could indicate the damage level at one or more critical points.

Based on the schematic diagram, an open-loop control policy was formulated [39]. An allowable damage rate level is set, when $v(t)$ is bigger than this level, LEC is activated to produce an additional control input to the system to make the damage rate return to within that set level. Further work on life extending control has been reported by Dai and Ray (1996) [10], Kallappa et al. (1997) [23], and Holmes and Ray (1998) [21] for different applications including rocket engines and fossil power plants in a framework of feedforward and feedback control.

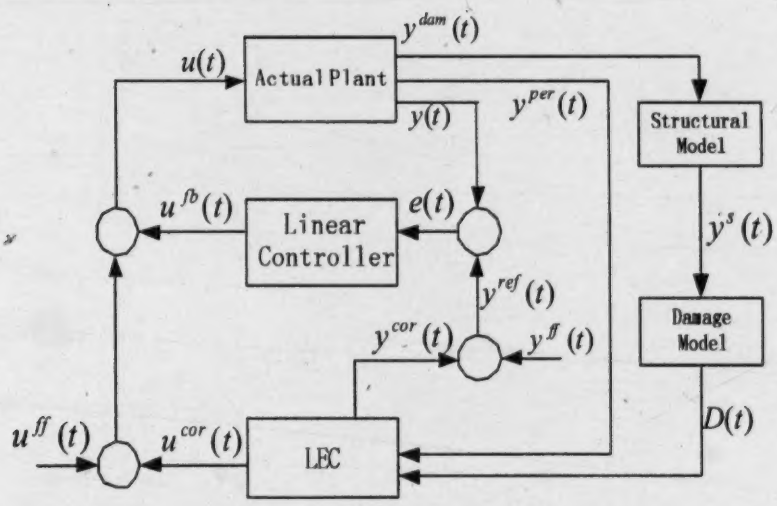

Figure 1.2: Another schematic diagram of LEC

Holmes (1997) [20] gave another schematic diagram of LEC in his Ph.D thesis shown in Fig. 1.2. Holmes's structure can reflect the main idea of LEC more clearly than Ray's. In Fig. 1.2, the outputs of plant are divided into three sets, denoted y^{dam}, y^{per} and y. Here $y^{dam}(t)$ represents those plant outputs necessary for the structural model to calculate the input vector of the damage model. For the example of fatigue damage in the turbine blades of a rocket engine or an aircraft engine, $y^{dam}(t)$ may be a vector whose elements contain the inlet gas temperature and the torque applied to the turbine shaft, and the structural model output, $y^s(k)$, may consist of stresses in the turbine blades. The outputs of the structural model are used by the damage model to produce information related to the damage in the critical component, $D(k)$. The variable $D(k)$ may contain information such as accumulated damage or damage rate. The plant's second output vector, $y^{per}(t)$, represents a measure of plant performance, and is the set of inputs to the damage controller. The combination of sequence of samples of the plant's is the third output vector, $y(t)$.

The outputs of the structural model are used by the damage model to produce information related to the damage in the critical component, $D(t)$. The variable $D(t)$ may contain information such as accumulated damage or damage rate. The purpose of the damage controller is to regulate the tradeoff between plant performance and damage rate and accumulation in critical plant components by applying corrections, $y^{cor}(t)$, to the desired setpoint, $y^{ff}(t)$. A possible second output vector of the damage controller is $u^{cor}(t)$, which

is a correction immediately added to the plant input signal. The sequence $y^{II}(t)$ is either chosen on-line by the human operator or, if the desired maneuver is known a priori, it can be determined in an optimal manner, along with the feedforward input sequence, $u^{II}(t)$, using off-line optimization. The purpose of the linear controller is to use the error sequence, $e(t)$, to provide feedback corrections, $u^{fb}(t)$, to the feedforward plant input, $u^{II}(t)$.

Zhang and Ray (1998) [55] have demonstrated the efficacy of LEC on a laboratory test apparatus where peak stresses in a critical component are reduced to increase its structural durability.

There are two main topics in LEC: fatigue damage modeling and life extending controller design. Before Ray and his co-workers [38, 39] proposed their modeling theory in 1994, most fatigue damage models were cycle-dependent, which were not suitable for on-line control implementation. However, the application of continuum models based on Ray's theory is hindered by the requirement of identifying the cycle extrema, a reference point.

Because the fatigue analysis is based on the Palmger-Miner (PM) theory [31] which is a cycle-dependent, it is necessary to reduce the complex history into a number of events which can be compared to the available constant amplitude test data. This process is called as cycle-counting. There are many cycle-counting methods, for example, level-crossing counting , peak counting, and simple-range counting [3]. But all these cycle counting methods described above make no consideration of the order in which cycles are applied. Due to the nonlinear relationship between stress and strain (plastic material behavior), the order in which cycles are applied has a significant effect on the stress-stain response of a material to applied loading. There are many examples to show two histories yield identical cycle counts when using the level-crossing and peak cycle counting methods, but the cyclic stress-strain responses of the two histories are recognized as being very different fatigue lives. Therefore, any fatigue analysis technique should account for these differences. Failure of early cycle counting methods to predict the fatigue life of components subjected to variable amplitude histories has in part been attributed to the fact that these sequence effects have not been included. It is this reason for that the Rainflow method has found a wide application.

The original Rainflow method of cycle-counting derived its name from an analogy used by Matsuishi and Endo [3] in their early work on this subject. At present a number of Rainflow cycle-counting techniques are in use, the most common being the original Rainflow method, range-pair counting, hysteresis loop counting, the " racetrack" method, ordered overall range counting, ranged-pair-range counting, and the Hayes method [3]. If the

strain-time history being analyzed begins and ends at the strain value having the largest magnitude, whether it occurs at a peak or a valley, all of these methods above yield identical results. However, if an intermediate strain value is used as a starting point, these methods will yield results that are similar but not identical in all cases. To know which point has the maximum magnitude, before counting cycles, it is necessary to know the whole strain history. This makes modeling methods, including the use of Rainflow cycle-counting, unsuitable for use in on-line control. An improved Rainflow cycle-counting method which was suggested by Downing in 1982 [13] can be used to overcome this drawback. This improved Rainflow cycle-counting method, called as 'one pass' Rainflow cycle-counting algorithm, overcomes this limitation and identifies the same cycles as the traditional Rainflow cycle-counting methods starting with a largest magnitude stress.

Ray's model with the improved Rainflow cycle-counting method is suitable for on-line control, however clearly, for control system design it is desirable to create as simple a continuum model as possible. For this reason, we also try a new modeling method suggested by Lorezo [25].

Besides modeling, life extending controller design is another important topic in LEC. As in traditional controller design, the difficulties to design an LEC for a boiler system are mainly from the two facts: first, a boiler system includes multi-inputs and multi-outputs, and interactions exist among different inputs and outputs, i.e. it is a multivariable system; second, there exist constraints, such as physical limits of actuators, safety margins, limited manufacture tolerances.

Conventional control design for boiler systems generally applies PID control theory. However, the PID method in general does not handle multivariable systems well; moreover, it cannot deal with constraints on inputs and outputs directly. Hence many PID controllers designed for boiler systems are either too conservative – operating far away from constraints – or too aggressive – violating these constraints. Both cases are reasons for under-performed control systems.

In order to cope with multivariable and constrained control problems, in recent years model predictive control (MPC) has been investigated and successfully employed. The main idea of MPC is to use a model of the plant to produce the future evolution of the system, and accordingly, select the command input. Prediction thus are handled according to the so called receding horizon scheme [4].

MPC, also known as Receding Horizon method, is the only advanced control technique to have had a significant and widespread impact on industrial process control. The current

interest of the processing industry in MPC can be traced back to a set of papers which appeared in the late 1970s. In 1978 Richalet et al [41] described successful applications of "Model Predictive Heuristic Control" and in 1980 Culter and Ramaker [9] outlined " Dynamic Matrix Control" (DMC) and reported applications to a fluid catalytic cracker. In both algorithms an explicit dynamic model of a plant is used to predict the effect of future actions of the manipulated variables on the output (thus the name "Model Predictive Control"). The future moves of the manipulated variables are determined by optimization with the objective of minimizing the predicted error subject to operating constraints. The optimization is repeated at each sampling time based on updated information from the plant. Since the rediscovery of MPC in 1978 and 1980, its popularity in the Chemical Process Industries has increased steadily [18]. Mehra et al. (1982) [30] reviewed a number of applications including a superheater, a steam generator, a wind tunnel, a utility boiler connected to a distillation column and a glass furnace. Shell [18] has applied MPC to many systems, such as a fluid catalytic cracking unit and a highly non-linear batch reactor.

The main reasons for the success of MPC are [28]:

1. It handles multivariable control problems naturally;
2. It can take account of actuator limitations;
3. It allows operation closer to constraints (compared with conventional control), which frequently leads to more profitable operation. It remarkably shorten pay-back periods;
4. Control update rates are low in most process industry, so that there is plenty of time for the necessary on-line computation.

It is difficult to identify the first person who proposed MPC theory. In fact, some practitioners in industry had implemented predictive control several years before the first publication on MPC even appeared. However, we try to give an account based on the published work.

Culter and Remaker [6] proposed a kind of predict control algorithm, called Dynamic Matrix Control (DMC). This emphasized optimal plant operation under constraints, and computed the control signal by repeatedly solving a linear programming (LP) problem. DMC went on to become the most well-known commercial MPC products. Following that a few varieties of MPC were suggested by different researchers later, e.g.,

MPHC	Richalet et al. (1977) [41].
MAC	Mehra et al. (1979) [30].
MOCCA	Sripada and Fisher (1980) [47].
EHPC	Ydstie et al. (1985) [54].
IMC	Garcia and Morari (1982) [18].
GPC	Clarke et al. (1984) [8].
MGPC	Shah, Mohtadi & Clarke (1987) [44].

1.4 Our research

In our research, a boiler-turbine system in an industrial co-generation system at Syncrude Canada's Utility plant is the object of our study. This system has three inputs: the feedwater flow (u_1), fuel flow (u_2) and attemperator spray flow (u_3), and three outputs: the drum level (y_1), drum pressure (y_2) and steam temperature (y_3). Because of physical limits, there are constraints on inputs and outputs. On the other hand, the superheater tubes which connect with the boiler are quite sensitive to peak temperature, so the steam temperature should not have big overshoot, we consider this requirement also.

In many industrial cases such as the boiler-turbine system at hand, control systems are already in operation for good dynamic performance. An important and practical question is: How to enhance the structural durability with existing control systems still in place? Fig. 1.3 is our proposed structure for LEC, in which the existing controller is kept in place for the plant, but a damage model and LEC are introduced on top at a higher level, which

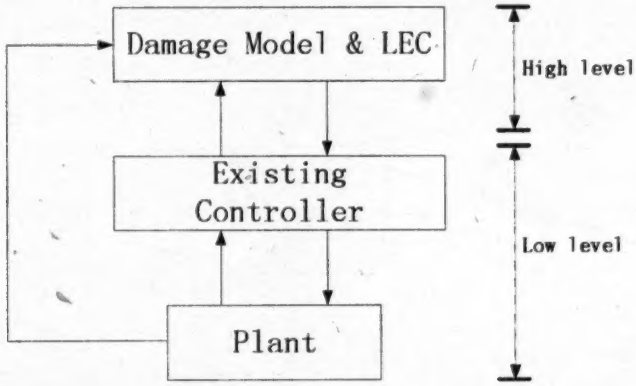

Figure 1.3: Combining LEC and existing controller

interact with both the existing controller and the plant. The damage models are used to estimate the accumulated damage and the damage rate within the system, based on measurable variables; the LEC is used to maintain the accumulated damage and damage

rate within prescribed limits, while trading off little dynamic performance. This way we gain improved component durability and longer service life without affecting much of the routine operation.

Incorporating the above idea, we propose a new LEC strategy shown in Fig. 1.4, where

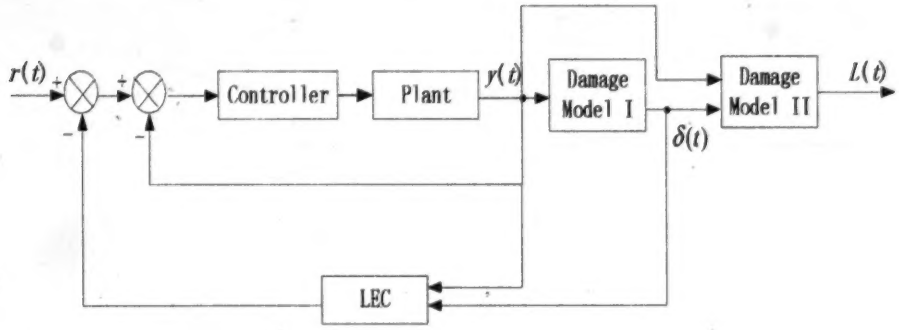

Figure 1.4: A new LEC closed-loop scheme setup

the inner feedback loop represents the existing control system, designed for system dynamic performance, damage models I and II, together with the LEC in the outer feedback loop, are introduced new. For the signals involved in Fig. 1.4, $r(t)$ is the reference input, $y(t)$ the plant output, $\delta(t)$ the damage variable (used for LEC feedback), and $L(t)$ the output of damage model II for life prediction. Note that the controller is kept in place, and no changes are necessary; the LEC is introduced for reducing damage level of critical components in the system. Damage model I is used for on-line LEC feedback; while damage model II for life prediction. Both models should be suitable for on-line implementation. This control structure has the following features:

- *Hierarchical*: The LEC is at a higher level with the existing controller in place; it provides an additional control signal to enhance durability and safety within the system. (This would be welcomed by practitioners and operators for ease in implementing the LEC.)

- *Good dynamic performance*: The LEC is designed to have minimum effect on the dynamic performance already achieved by the existing control system.

- *Optimal tradeoff*: We target optimal tradeoff between component durability and dynamic performance.

In our research, both main topics in LEC are related. In damage modeling, we try to

build a model which is suitable for on-line control implementation; in LEC design, linear quadratic (LQ) and MPC theory are applied.

The system to be studied is a boiler system in an industrial co-generation system at Syncrude Canada's utility plant. We have a sophisticated nonlinear model developed, the SYNSIM model [42]. This model has been extensively tested, and its predictions are consistent with measurements from the real plant. But it is not practical for controller design based on this sophisticated nonlinear model. Instead we design our LEC with a linear model which is linearized from the SYNSIM model at a normal operation condition. Then simulation is conducted using both the linear and SYNSIM models. Because the formulation of LEC fit the frame of the LQ theory very well, before proceeding MPC design, we use LQ to design an LEC first.

In Chapter 2, we discuss two continuum fatigue modeling theories and tailor them into our problem. In Chapter 3, the principle and structure of the controlled plant are introduced briefly. In chapter 4, an LEC is designed using LQ theory and in Chapter 5, we concentrate on designing an LEC with MPC theory and in both chapters (chapter 4 and chapter 5) simulations are conducted with both the linear model and SYNSIM model.

Chapter 2

Fatigue Damage Modeling

2.1 The basis of fatigue damage modeling: Cycle-dependent modeling

In the literature, one can look up the stress-fatigue life curves (S-N curves) of many materials. S-N curves are the basis for fatigue modeling. Fig. 2.1 is an example of an S-N curve, if

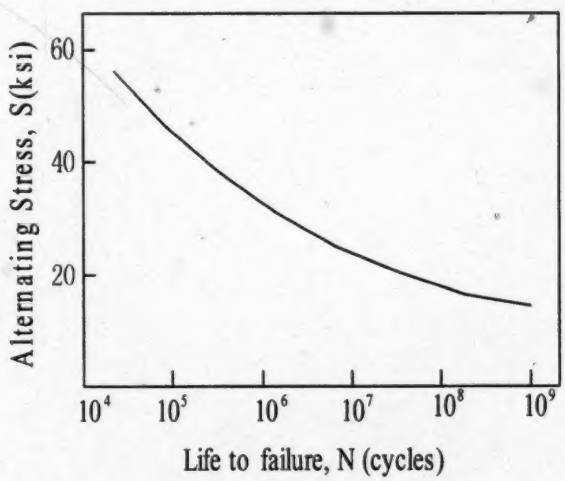

Figure 2.1: S-N curve for some material

a stress with the level S_i is applied, the corresponding life to failure, N_{fi}, can be found from this figure. Most experimental data are generated by applying constant amplitude loads to test specimens. However, real components always work under fluctuated loads, so the task of a model is to predict the life of a component subjected to varying amplitude loads using constant amplitude loads test data. Palmgren in 1924 proposed a linear damage rule which

was further developed by Miner in 1945 [3]. This Palmgren-Miner (P-M) method can be described by the following formula

$$D = \sum_i D_i = \sum_i \frac{n_i}{N_i}, \tag{2.1}$$

where D is the damage variable (cumulative damage), n_i is the number of load cycles at the (constant) stress level σ_i, and N_i is the total number of cycles to failure at the same stress level. Although the P-M method gives only approximate estimation of damage, it has found many applications because of its simplicity. The main disadvantage of the P-M method is that it doesn't account for the sequential effect [3]. This theory predicts that the damage caused by a stress cycle is independent of where it occurs in the load history.

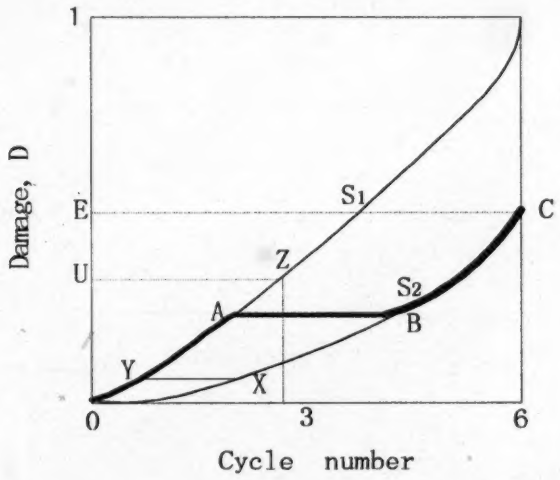

Figure 2.2: Demonstration of nonlinear fatigue damage modeling

To improve the P-M rule, many nonlinear cumulative damage rules have been proposed such as those by Gatts, Dolm, Martin, and Manson. One nonlinear damage approach, which was proposed by Richard and Newmark and further developed by Marco and Starkey [3], is given by

$$D = \sum_i D_i = \sum_i \left(\frac{n_i}{N_i}\right)^{P_i}, \tag{2.2}$$

where the exponent, P_i, is a function of stress level σ_i. This approach is of current research interest and has applications in design. Although nonlinear methods may give better predictions than Miner's rule for two-step histories, it cannot be guaranteed that they will work better for actual service load histories [3]. The application of this method is shown in Fig. 2.2 which is a plot of damage fraction versus cycle for two stress level, where $S_1 > S_2$,

intuitively, the damage should be bigger when S_1 is applied first than that when S_2 is applied first. Now we assume both S_1 and S_2 are applied for two cycles. In Fig. 2.2, the path $O-A-B-C$ represents a high-low stress history while the path $O-X-Y-Z$ a low-high stress history. From this figure, we see that when S_1 is applied first, the damage is E which is bigger than U, the damage when S_2 is applied first.

From the above, we see that both linear and nonlinear models are cycle-dependent. But cycle-dependent models are not suitable for on-line control implementation. Ray in 1994 [38] proposed a continuous-time fatigue model which is discussed in the next section.

2.2 Ray's modeling method

2.2.1 Ray's method

Consider a critical component subject to cyclic loading; let ϵ_r be the total strain corresponding to the reference stress σ_r at the starting point R of a given reversal as determined from the Rainflow cycle-counting method, and ϵ and σ be the strain and stress of the point on the same reversal with R respectively. We define

$$\begin{aligned} \triangle\epsilon &= |\epsilon - \epsilon_r|, \\ \triangle\sigma &= |\sigma - \sigma_r|. \end{aligned}$$

The relationship between the stress σ and the strain ϵ can be described by the following equation: [15]

$$\frac{\triangle\epsilon}{2} = \frac{\triangle\sigma}{2E} + (\frac{\triangle\sigma}{2K'})^{\frac{1}{n'}}, \tag{2.3}$$

where E is the modulus of elasticity, K' the cyclic strength coefficient and n' the cyclic strain-hardening coefficient, all determined experimentally based on material properties. Note that $\frac{\triangle\epsilon}{2}$ and $\frac{\triangle\sigma}{2}$ are stress and strain amplitudes on the critical point of the component, respectively. And the relationship of strain and stress also can be shown in Fig. 2.3 , where total strain ϵ_t can be divided into two parts, the elastic strain ϵ_e and the plastic strain ϵ_p. The elastic strain ϵ_e is the part which can be recovered when unloading. Break the right-hand side of Eq. (2.3) into two parts – the first term corresponds to the elastic part (ϵ_e) of the strain and the second the plastic part (ϵ_p). Thus we have

$$\frac{\triangle\epsilon_e}{2} = \frac{\triangle\sigma}{2E}, \tag{2.4}$$

$$\frac{\triangle\epsilon_p}{2} = (\frac{\triangle\sigma}{2K'})^{\frac{1}{n'}}. \tag{2.5}$$

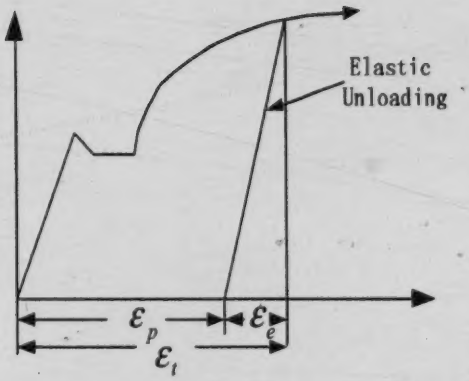

Figure 2.3: Elastic and plastic strains

Assume that the life cycle of this component is N; the following formula by Dowling [12] can be used to predict the component life:

$$\frac{\triangle\epsilon}{2} = \frac{\sigma_f^{'}}{E}(1-\frac{\sigma_m}{\sigma_f^{'}})(2N)^b + \epsilon_f^{'}(1-\frac{\sigma_m}{\sigma_f^{'}})^{\frac{c}{b}}(2N)^c.$$

Here $\sigma_f^{'}$ is the fatigue strength coefficient, $\epsilon_f^{'}$ the fatigue ductility coefficient, σ_m the mean stress, b the fatigue strength exponent, and c the fatigue ductility exponent. Replacing N by $\frac{1}{\triangle\delta}$, where $\triangle\delta$ represents the increment in linear damage δ during one cycle, we have

$$\frac{\triangle\epsilon}{2} = \frac{\sigma_f^{'}}{E}(1-\frac{\sigma_m}{\sigma_f^{'}})(\frac{\triangle\delta_e}{2})^{-b} + \epsilon_f^{'}(1-\frac{\sigma_m}{\sigma_f^{'}})^{\frac{c}{b}}(\frac{\triangle\delta_p}{2})^{-c}, \tag{2.6}$$

δ_e and δ_p being the elastic and plastic damage increments, respectively. Similarly, we break the right-hand side of (2.6) into two parts:

$$\frac{\triangle\epsilon_e}{2} = \frac{\sigma_f^{'}}{E}(1-\frac{\sigma_m}{\sigma_f^{'}})(\frac{\triangle\delta_e}{2})^{-b}, \tag{2.7}$$

$$\frac{\triangle\epsilon_p}{2} = \epsilon_f^{'}(1-\frac{\sigma_m}{\sigma_f^{'}})^{\frac{c}{b}}(\frac{\triangle\delta_p}{2})^{-c}. \tag{2.8}$$

From (2.4) and (2.7), $\triangle\delta_e$ can be calculated as follows:

$$\triangle\delta_e = 2\times(\frac{\triangle\sigma}{2(\sigma_f^{'}-\sigma_m)})^{-\frac{1}{b}}.$$

From (2.5) and (2.8), $\triangle\delta_p$ can be obtained by

$$\triangle\delta_p = 2\times(\frac{1}{\epsilon_f^{'}}(\frac{\triangle\sigma}{2K^{'}})^{\frac{1}{n^{'}}}\times(1-\frac{\sigma_m}{\sigma_f^{'}})^{-\frac{c}{b}})^{-\frac{1}{c}}.$$

According to the Rainflow method, σ_r is either a local minimum or maximum; so $\frac{d\sigma_r}{dt} = 0$. Further, since it is assumed that no damage occurs during unloading, the damage rate can be made equal to zero when $\sigma < \sigma_r$. When $\sigma \geq \sigma_r$, we have

$$\begin{aligned}
\frac{d\delta_e}{dt} &= 2 \times \frac{d}{d\sigma}((\frac{\sigma - \sigma_r}{2(\sigma_f' - \sigma_m)})^{-\frac{1}{b}}) \times \frac{d\sigma}{dt}, \\
\frac{d\delta_p}{dt} &= 2 \times \frac{d}{d\sigma}((\frac{1}{\epsilon_f'} \times (\frac{\sigma - \sigma_r}{2K'})^{\frac{1}{n'}}) \times (1 - \frac{\sigma_m}{\sigma_f'})^{-\frac{c}{b}})^{-\frac{1}{c}}) \times \frac{d\sigma}{dt}.
\end{aligned}$$

Here $\frac{d\sigma}{dt}$ can be found from direct measurements of the strain rate by specific gauges or by the Finite Element Analysis (FEA). When both are not available, Lu [26] suggested an alternate formula:

$$\sigma(\tau) = \frac{E\alpha}{1-\nu}[T_m(\tau) - T(\tau)],$$

with ν the Poisson's ratio, α the coefficient of linear expansion, T the temperature of the component at the critical point, and T_m the mean temperature. Thus the total damage rate is the weighted sum of the elastic damage rate and the plastic damage rate:

$$\frac{d\delta}{dt} = w\frac{d\delta_e}{dt} + (1-w)\frac{d\delta_p}{dt}.$$

The weighting function, w, is based on the elastic and plastic strain amplitudes

$$w = \frac{\epsilon_e - \epsilon_{re}}{\epsilon - \epsilon_r} = \frac{\Delta\epsilon_e}{\Delta\epsilon},$$

where ϵ_{re}, the elastic part of ϵ_r, is defined as

$$\epsilon_{re} = \frac{\sigma_r}{E}.$$

Therefore, the accumulated damage D can be found by

$$D = \sum_{k=1}^{N} \delta(k).$$

D is the fraction of life used up by an event or a series of events. Failure in any of the cumulative damage theories is assumed to occur when the summation of damage equals 1.

2.2.2 Rainflow cycle-counting

We notice that this continuous-time model involves the use of the Rainflow cycle-counting method to specify the starting point of one cycle to be the reference point R. To show the advantages and disadvantages of Ray's modeling theory, it is necessary to understand the details of the Rainflow method. The Rainflow method of cycle-counting derived its name from an analogy used by Matsuishi and Endo in their early work on this subject.

The rules specifying the manner in which rain falls are as follows [3]:

1. To eliminate the counting of half cycles, the strain-time history is drawn so as to begin and end at the strain value of greatest magnitude.

2. A flow of rain is begun at each strain reversal in the history and is allowed to continue to flow unless;

   (a) The rain begins at a local maximum point (peak) and falls opposite a local maximum point greater than that from which it came.

   (b) The rain began at a local minimum point (valley) and falls opposite a local minimum point greater (in magnitude) than that from which it came.

   (c) It encounters a previous rainflow.

Rainflow cycle-counting (ASTM standard). While Rainflow cycle-counting can be completed manually for relatively simple load histories, for more complex loading histories the method is better implemented through the use of computers. In 1986, American Society for Testing and Materials (ASTM) in its Annual Book of ASTM Standards [1] gave several Rainflow cycle-counting algorithms which can easily be adapted into computer programs:

1. Read the next peak or valley. If out of data, stop.

2. If there are less than three points, go to step 1. For ranges X and Y using the three most recent peaks and valleys that have not been discarded. (X: range under consideration; Y: previous range adjacent to X.)

3. Compare the absolute values of ranges X and Y:

   (a) If $X < Y$, go to step 1.

   (b) If $X \geq Y$, go to step 4.

4. Count range Y as one cycle; discard the peak and valley of Y; and go to step 2.

Now we use an example to clarify the work of the foregoing procedure. Fig. 2.4 shows a strain history. Because we have to start with the point C with the greatest amplitude, we preceed our counting with Fig. 2.5. Fig. 2.6 to Fig. 2.18 show the counting process. In Fig. 2.6, the first peak has been read into the vector. Since there are less than 3 points in the vector, ranges X and Y are undetermined and the next peak or valley must be read. In Fig. 2.7, point D has been read into the vector and the stress is unloaded from C to D along the outer loop curve. Range Y is still undetermined so the next peak or valley must

be read. In Fig. 2.8, point E has been read and the stress increases from D to E along the outer loop curve. A closed hysteresis loop has been formed and, according to the counting rules, range Y should be counted and its points discarded since they have no bearing on future events. The counting algorithm identified the same cycle, DC, as was determined from the stress/strain response. In Fig. 2.9, points D and C have been eliminated from the contents of the vector. It is easy to follow Fig. 2.10-Fig. 2.18 along with the counting rules to see that the algorithm identifies the same cycles (DC, GF, BA, HE) as were determined from the stress/strain response.

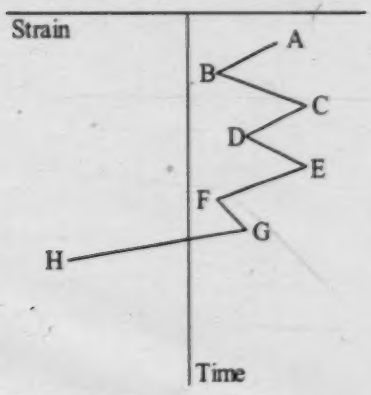

Figure 2.4: Variable amplitude strain history

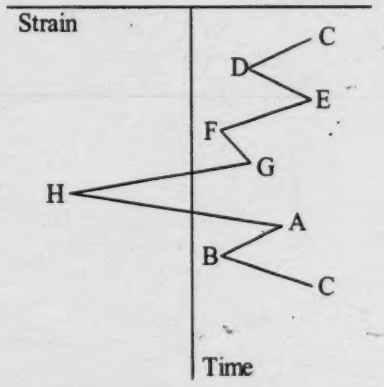

Figure 2.5: Rearrangement of the strain history

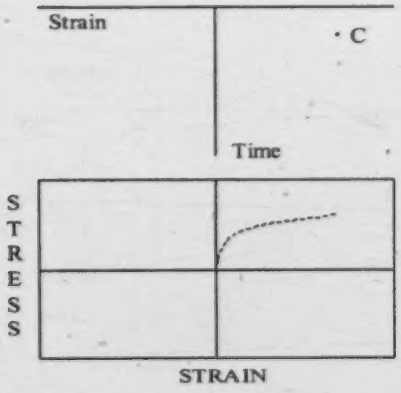

Figure 2.6: Step 1 of Rainflow cycle-counting

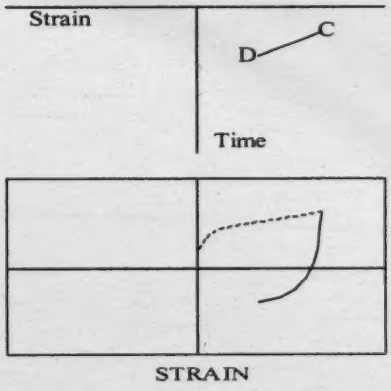

Figure 2.7: Step 2 of Rainflow cycle-counting

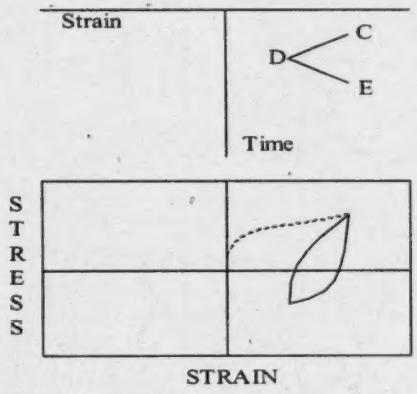

Figure 2.8: Step 3 of Rainflow cycle-counting

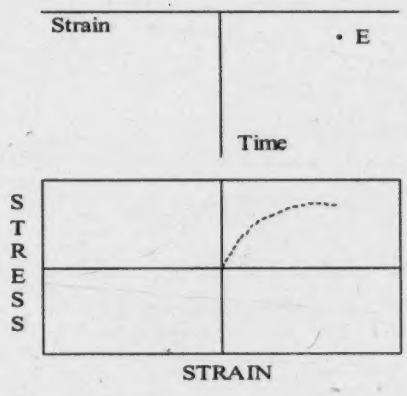

Figure 2.9: Step 4 of Rainflow cycle-counting

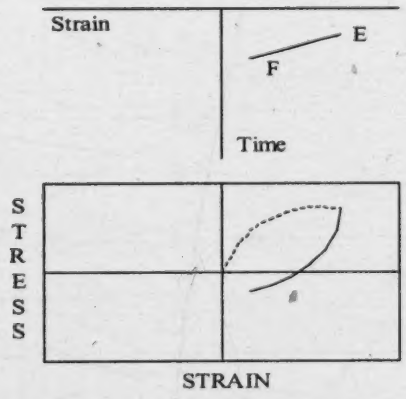

Figure 2.10: Step 5 of Rainflow cycle-counting

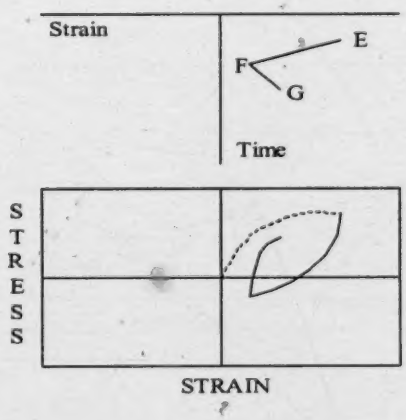

Figure 2.11: Step 6 of Rainflow cycle-counting

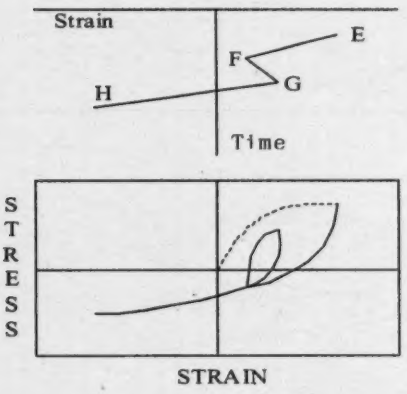

Figure 2.12: Step 7 of Rainflow cycle-counting

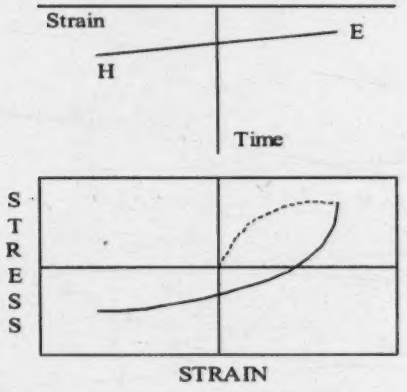

Figure 2.13: Step 8 of Rainflow cycle-counting

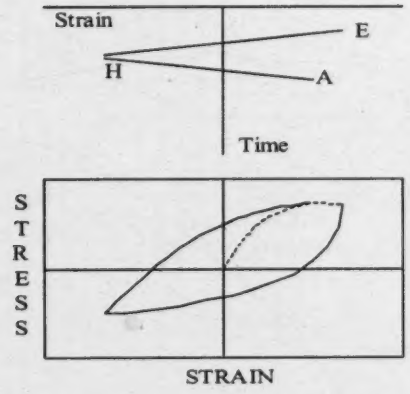

Figure 2.14: Step 9 of Rainflow cycle-counting

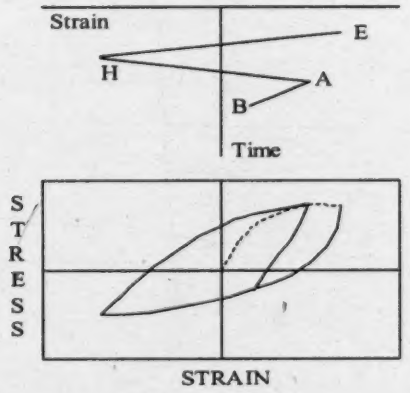

Figure 2.15: Step 10 of Rainflow cycle-counting

2.2.3 Improved Rainflow cycle-counting

Several algorithms [2] are available to perform the counting; however, the start point should be at the strain value of greatest magnitude. Thus they all require that the entire load history be known before the counting process starts. As a result, they are not suitable for on-line control implementation. For this reason, this continuous-time model we discussed is not suitable for on-line control [48].

An improved Rainflow cycle-counting method, called the "one pass" method, was proposed by Downing [13], which allows taking any point as a temporal start point during the counting process; if a certain condition is satisfied, the starting point should be moved to the next peak. This improved method can give the same result as the conventional Rainflow method with the added advantage of causal implementation. Thus, incorporating this method in Ray's continuous model allows on-line implementation of life extending controllers. This Rainflow algorithm counts a history of peaks and valleys in sequence as they occur. It calculates the same ranges and means as other Rainflow Algorithms which require that the history be rearranged to begin and end with the maximum peak (or minimum valley). This improved Rainflow cycle-counting then proceeds according to the following steps:

1. Read the next peak or valley (if out of data, go to step 6).
2. Form ranges X and Y (If the vector contains less than 2 points, past the starting point, go to step 1).

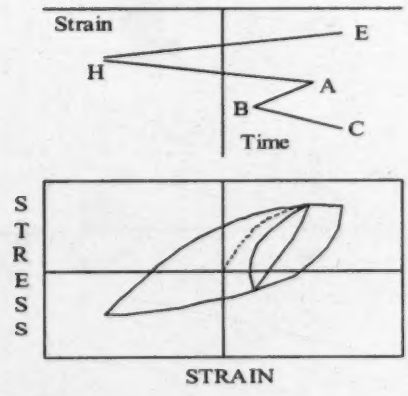

Figure 2.16: Step 11 of Rainflow cycle-counting

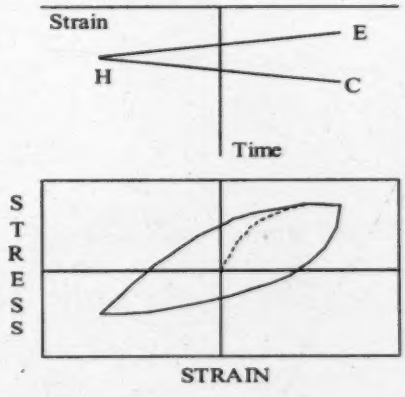

Figure 2.17: Step 12 of Rainflow cycle-counting

3. Compare ranges X and Y:

   (a) If $X < Y$, go to step 1;

   (b) If $X = Y$ and Y contains S, go to step 1;

   (c) If $X > Y$ and Y contains S, go to step 4;

   (d) If $X \geq Y$ and Y does not contain S, go to step 5.

4. Move S to the next point in the vector, go to step 1.

5. Count range Y. Discard the peak and valley of Y, and go to step 2.

6. Read the next peak or valley from the beginning of the vector $E(n)$. (If the starting point, S, has already been reread, STOP.)

7. Form ranges X and Y. (If the vector contains less than 2 points past the starting point, go to step 6.)

8. Compare ranges X and Y:

   (a) If $X < Y$, go to step 6;

   (b) If $X \geq Y$, go to step 9;

9. Count range Y. Discard the peak and valley of Y. Go to step 7.

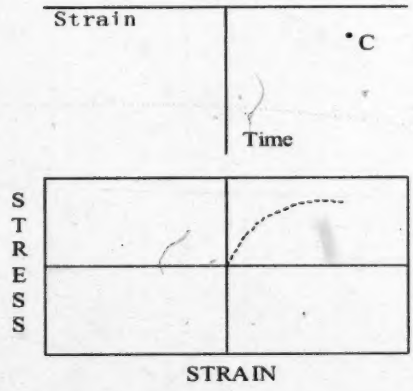

Figure 2.18: Step 13 of Rainflow cycle-counting

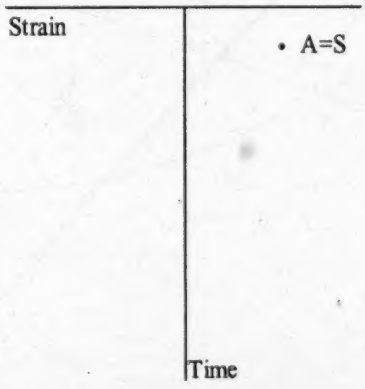

Figure 2.19: Step 1 of the improved Rainflow cycle-counting

The same example as the last one is still used to show the process of the improved Rainflow cycle-counting method, one-pass method. Fig. 2.19 to Fig. 2.33 show the counting

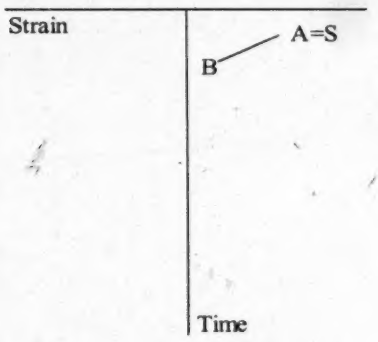

Figure 2.20: Step 2 of the improved Rainflow cycle-counting

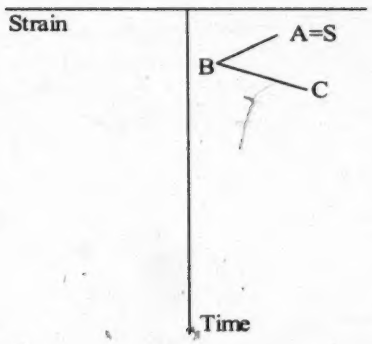

Figure 2.21: Step 3 of the improved Rainflow cycle-counting

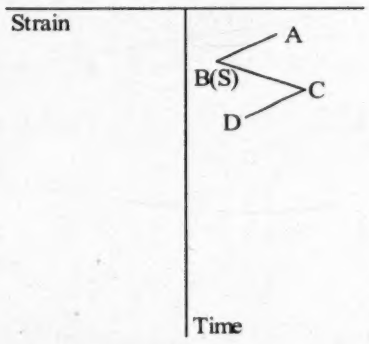

Figure 2.22: Step 4 of the improved Rainflow cycle-counting

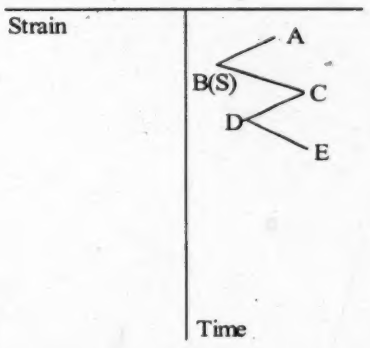

Figure 2.23: Step 5 of the improved Rainflow cycle-counting

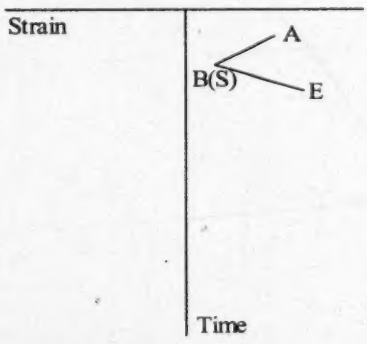

Figure 2.24: Step 6 of the improved Rainflow cycle-counting

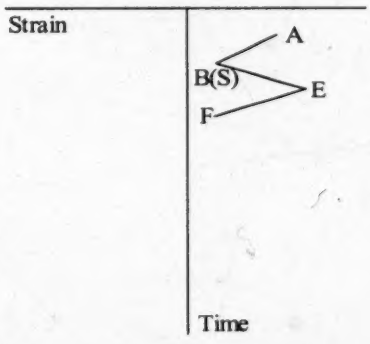

Figure 2.25: Step 7 of the improved Rainflow cycle-counting

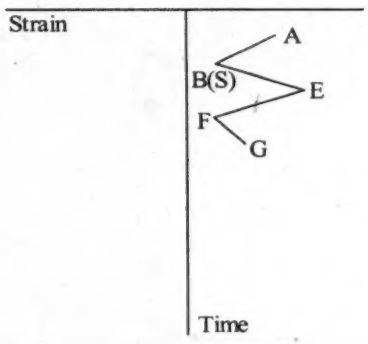

Figure 2.26: Step 8 of the improved Rainflow cycle-counting

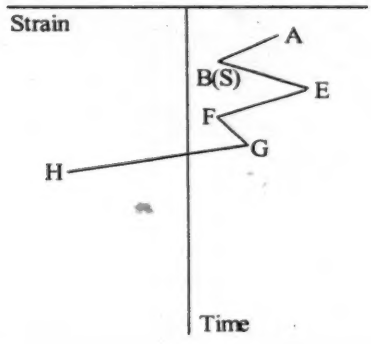

Figure 2.27: Step 9 of the improved Rainflow cycle-counting

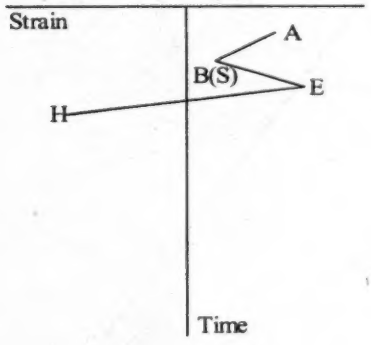

Figure 2.28: Step 10 of the improved Rainflow cycle-counting

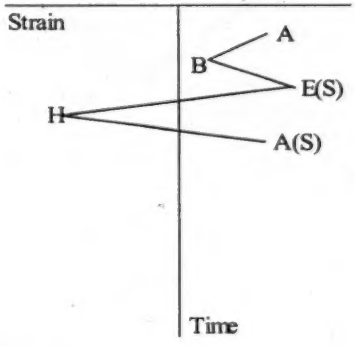

Figure 2.29: Step 11 of the improved Rainflow cycle-counting

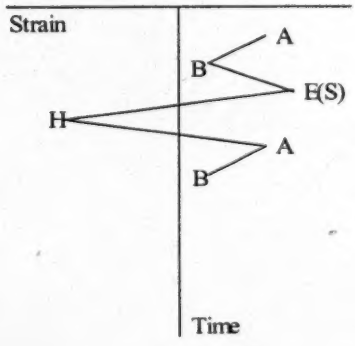

Figure 2.30: Step 12 of the improved Rainflow cycle-counting

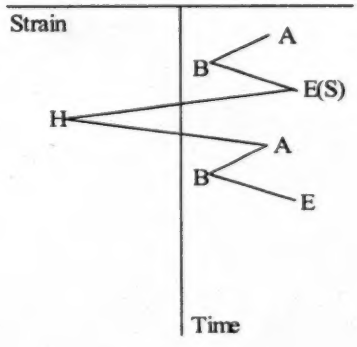

Figure 2.31: Step 13 of the improved Rainflow cycle-counting

decisions for each step in the counting process. It should be noted that the starting point, S, is always the first occurrence of peak or valley. In figure 2.19, we begin our counting with the first peak A; since there are less than 3 points in the vector, ranges X and Y are undetermined and the next peak or valley must be read in. In Fig. 2.20, point B has been read into the vector. Range Y is still undetermined so the next peak or valley must be read. In Fig. 2.21, point C has been read. Now X and Y can be formed, then we have $X > Y$ and Y contains point S, so the starting point S should be moved to the next valley B and the next peak or valley must be read. In Fig. 2.22, point D has been read and S becomes point B, we have $X < Y$, so the next peak or valley should be read. In Fig. 2.23, point E has been read. We have $X = Y$ and Y does not contain S, according to the counting rules, range Y should be counted and its points discarded since they have no bearing on future events. It is easy to understand the process of counting from Fig. 2.24 to Fig. 2.33. Through Fig. 2.19 to Fig. 2.33, the same cycles (DC, GF, BA, HE) have been identified as in the previous algorithm. The important fact is that the improved method identifies the same cycles as the previous algorithm without the restriction that history be rearranged to begin and end with the maximum peak.

2.3 Lorenzo's modeling method

From the analysis above, we know that Ray's model is hindered by the requirement of identifying the cycle extrema and then calculating the damage between the extrema based on extrema information. We can apply the improved Rainflow cycle-counting method to overcome this drawback. But clearly, for a control system design it is desirable to create

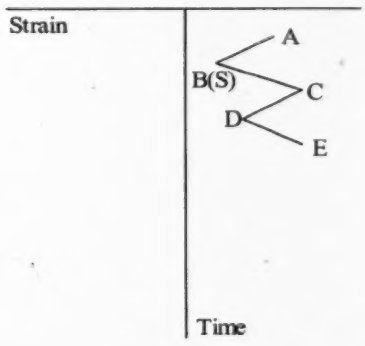

Figure 2.32: Step 14 of the improved Rainflow cycle-counting

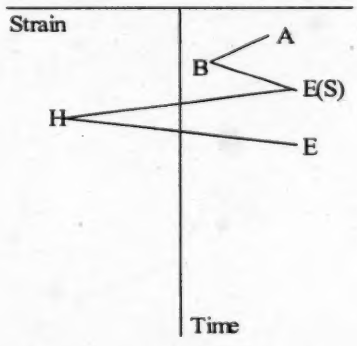

Figure 2.33: Step 15 of the improved Rainflow cycle-counting

as simple a continuum model as possible. For this reason, here we discuss a new modeling method suggested by Carl. F. Lorenzo [25].

2.3.1 Zero mean stress continuum damage model

We have seen that the following relationship exits:

$$\epsilon_a = \frac{\sigma_a}{E} + \left[\frac{\sigma_a}{A}\right]^{1/s}. \tag{2.9}$$

When zero mean stress, Dowling have shown that

$$\epsilon_a = \frac{\sigma_f^{'}}{E}(2N_f)^b + \epsilon_f^{'}(2N_f)^c. \tag{2.10}$$

The total strain amplitude is seen to be composed of two parts. The elastic strain contribution

$$\epsilon_{ae} = \frac{\sigma_a}{E} = \frac{\sigma_f^{'}}{E}(2N_f)^b \tag{2.11}$$

and the plastic strain contribution

$$\epsilon_{ap} = \left(\frac{\sigma_a}{A}\right)^{1/s} = \epsilon_f^{'}(2N_f)^c. \tag{2.12}$$

Now from Eq. (2.11), we have

$$\sigma_a = \sigma_f^{'}(2N_f)^b,$$

then

$$N_f = \frac{1}{2}\left(\frac{\sigma_a}{\sigma_f^{'}}\right)^{1/b}.$$

Substituting the above into Eq. (2.12)

$$\epsilon_{ap} = \epsilon_f^{'}\left(\frac{\sigma_a}{\sigma_f^{'}}\right)^{c/b}. \tag{2.13}$$

From the first part of Eq. (2.12)

$$\left(\frac{\sigma_a}{A}\right)^{1/s} = \left[\frac{\sigma_a}{\sigma_f^{'}}\left(\epsilon_f^{'}\right)^{b/c}\right]^{c/b}. \tag{2.14}$$

From the above equation, Lorenzo [25] deduced that the following relationships should hold:

$$\frac{1}{s} = \frac{c}{b} \tag{2.15}$$

and

$$A\epsilon_f^{'b/c} = A\epsilon_f^{'s} = \sigma_f^{'}. \tag{2.16}$$

Experiments show that Eq. (2.15) and Eq. (2.16) are approximately correct for many metallic materials. Combining Eq. (2.9) and Eq. (2.10)

$$\epsilon_a = \frac{\sigma_a}{E} + \left(\frac{\sigma_a}{A}\right)^{1/s} = \frac{\sigma_f^{'}}{E}(2N_f)^b + \epsilon_f^{'}(2N_f)^c . \tag{2.17}$$

Since $\delta_{cyc} = 1/N_f$, where δ_{cyc} is the damage per cycle this can be written as

$$\epsilon_a = \frac{\sigma_a}{E} + \left(\frac{\sigma_a}{A}\right)^{1/s} = \frac{\sigma_f^{'}}{E}\left(\frac{\delta_{cyc}}{2}\right)^{-b} + \epsilon_f^{'}\left(\frac{\delta_{cyc}}{2}\right)^{-c} . \tag{2.18}$$

Eq. (2.18) can be divided two parts and then are considered separately to determine δ_{cyc}. For the elastic part

$$\frac{\sigma_a}{E} = \frac{\sigma_f^{'}}{E}\left(\frac{\delta_{cyc_e}}{2}\right)^{-b} , \tag{2.19}$$

then

$$\delta_{cyc_e} = 2\left(\frac{\sigma_a}{\sigma_f^{'}}\right)^{-1/b} . \tag{2.20}$$

For the plastic part

$$\left(\frac{\sigma_a}{A}\right)^{1/s} = \epsilon_f^{'}\left(\frac{\delta_{cyc_p}}{2}\right)^{-c} , \tag{2.21}$$

yields

$$\delta_{cyc_p} = 2\left(\frac{1}{\epsilon_f^{'}}\right)^{-1/c}\left(\frac{\sigma_a}{A}\right)^{-1/cs} = 2\sigma_a^{-1/cs}\left(\frac{1}{A\epsilon_f^{'s}}\right)^{-1/cs} . \tag{2.22}$$

Using the relations of Eq. (2.15) and Eq. (2.16), we have

$$\delta_{cyc_p} = 2\left(\frac{\sigma_a}{\sigma_f^{'}}\right)^{-1/b} . \tag{2.23}$$

Comparing Eq. (2.20) and Eq. (2.23), we have that

$$\delta_{cyc} = 2\left(\frac{\sigma_a}{\sigma_f^{'}}\right)^{-1/b} \tag{2.24}$$

or

$$N_f = \frac{1}{2}\left(\frac{\sigma_a}{\sigma_f^{'}}\right)^{1/b} . \tag{2.25}$$

The above equation can be used in life prediction with a coarse precision. However, clearly it is cycle-dependent. We now extend this analysis towards a continuum damage model.

A typical stain-stress hysteresis loop is shown in Fig. 2.34. Now a critical question is: in Fig. 2.34, over which part of the cycle does the damage occur? Damage is not likely generated during the relaxation (unloading) phases of the cycle, namely $A \rightarrow B$ and

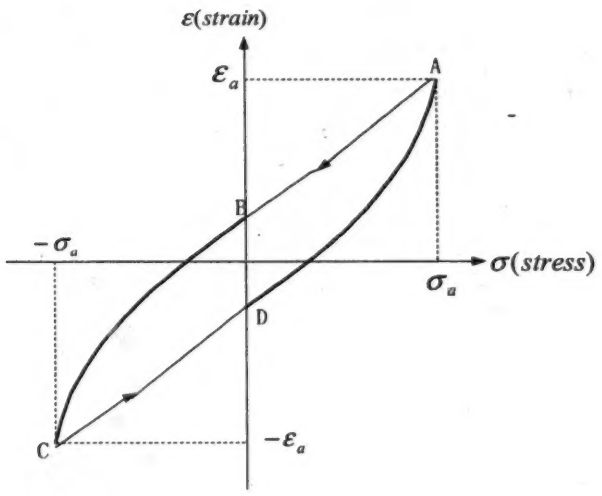

Figure 2.34: Strain-stress curve under zero mean stress

$C \to D$. While it is most likely during tensile stressing $D \to A$ and may also occur during compressive stressing $B \to C$.

From Eq. (2.24), we have

$$\int_{cycle} \delta'(\sigma) d\sigma = \delta_{cyc} = 2\left(\frac{\sigma_a}{\sigma_f'}\right)^{-1/b}. \tag{2.26}$$

Now we analysis the case for $\sigma > 0$, σ increasing, then

$$\int_0^{\sigma_a} \delta'(\sigma) d\sigma = \delta_{cyc} = 2\left(\frac{\sigma_a}{\sigma_f'}\right)^{-1/b}. \tag{2.27}$$

It is readily to show

$$\delta'(\sigma) = \frac{-2}{b\sigma_f'}\left(\frac{\sigma}{\sigma_f'}\right)^{-(1+b)/b}. \tag{2.28}$$

We also need to notice, when $\sigma < 0$, σ decreasing, compression stress occurs

$$\delta'(\sigma) = \frac{-2}{b\sigma_f'}\left(\frac{|\sigma|}{\sigma_f'}\right)^{-(1+b)/b} \quad \text{for } \sigma < 0, \sigma \text{ decreasing.} \tag{2.29}$$

In a word, we have

$$\delta'(\sigma) = \begin{cases} \frac{-2k}{b\sigma_f'}\left(\frac{|\sigma|}{\sigma_f'}\right)^{-(1+b)/b} & \text{for } \sigma > 0, \sigma \text{ increasing} \\ \frac{-2(1-k)}{b\sigma_f'}\left(\frac{|\sigma|}{\sigma_f'}\right)^{-(1+b)/b} & \text{for } \sigma < 0, \sigma \text{ decreasing} \end{cases} \tag{2.30}$$

where the parameter k weights the tension side damage relative to compressive side. In the time domain the damage rate $\dot{D}$ ($\sigma > 0$ and σ increasing) will be given as

$$\dot{D}(t) = \frac{d\delta}{d\sigma}\frac{d\sigma}{dt} = -\frac{2}{b\sigma_f'}\left(\frac{\sigma}{\sigma_f'}\right)^{-(1+b)/b}\frac{d\sigma}{dt} \tag{2.31}$$

and the accumulated damage will be

$$D(t) = \int_0^t -\frac{2}{b\sigma_f'}\left(\frac{\sigma}{\sigma_f'}\right)^{-(1+b)/b}\frac{d\sigma}{dt}dt. \tag{2.32}$$

2.3.2 Non-zero mean stress Continuum damage model

Mean stress is an important factor to evaluate damage [15], therefore we should consider the effect of the mean stress. Dowling also showed the following equation after including the mean stress σ_m

$$\epsilon_a = \frac{\sigma_f' - \sigma_m}{E}(2N_f)^b + \epsilon_f'\left(1 - \frac{\sigma_m}{\sigma_f'}\right)^{c/b}(2N_f)^c. \tag{2.33}$$

Assuming the cyclic stress-strain behavior of the material is not altered by mean stress then

$$\epsilon_a = \frac{\sigma_a}{E} + \left[\frac{\sigma_a}{A}\right]^{1/s}.$$

Similar with what we have done before

$$\frac{\sigma_a}{E} + \left[\frac{\sigma_a}{A}\right]^{1/s} = \frac{\sigma_f' - \sigma_m}{E}(2N_f)^b + \epsilon_f'\left(1 - \frac{\sigma_m}{\sigma_f'}\right)^{c/b}(2N_f)^c. \tag{2.34}$$

Equating the elastic part of the equation

$$\delta_{cyc_a} = 2\left(\frac{\sigma_a}{\sigma_f' - \sigma_m}\right)^{-1/b}. \tag{2.35}$$

For the plastic terms

$$\left(\frac{\sigma_a}{A}\right)^{1/s} = \epsilon_f'\left(1 - \frac{\sigma_m}{\sigma_f'}\right)^{c/b}\left(\frac{\delta_{cyc_p}}{2}\right)^{-c}, \tag{2.36}$$

yields

$$\delta_{cyc_p} = 2\left(\frac{\sigma_a}{\sigma_f' - \sigma_m}\right)^{-1/b}. \tag{2.37}$$

Comparing Eq. (2.35) and Eq. (2.37), we have

$$\delta_{cyc} = 2\left(\frac{\sigma_a}{\sigma_f' - \sigma_m}\right)^{-1/b}. \tag{2.38}$$

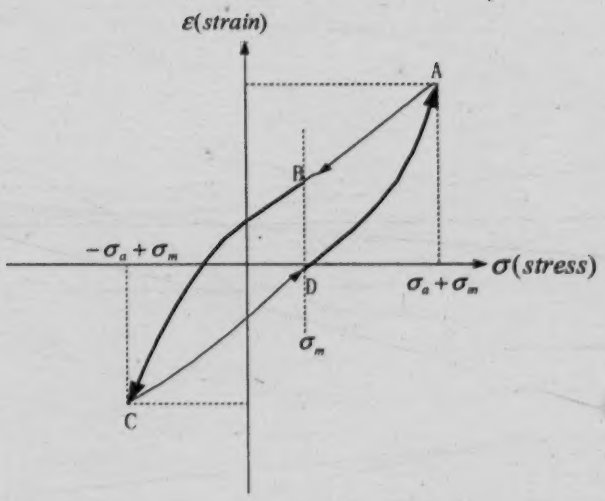

Figure 2.35: Strain-stress curve under non-zero mean stress

In Fig. 2.35, the case all cyclic damage occurs only between σ_m and $\sigma_m + \sigma_a$ on the tensile leg is considered. So

$$\int_{\sigma_m}^{\sigma_m+\sigma_a} \delta'(\sigma)d\sigma = \delta_{cyc} = 2\left(\frac{\sigma_a}{\sigma_f' - \sigma_m}\right)^{-1/b}. \tag{2.39}$$

We recall that

$$\int_a^x f(y)dy = \int_0^{x-a} f(y+a)dy, \tag{2.40}$$

then

$$\int_0^{\sigma_a} \delta'(\sigma+\sigma_m)d\sigma = 2\left(\frac{\sigma_a}{\sigma_f' - \sigma_m}\right)^{-1/b}. \tag{2.41}$$

Differentiating the above equation gives

$$\delta'(\sigma+\sigma_m) = -\frac{2}{b}\left(\frac{1}{\sigma_f' - \sigma_m}\right)^{-1/b}(\sigma)^{-(1/b)-1}. \tag{2.42}$$

The discussion for the compression mode ($\sigma < \sigma_m$, σ decreasing) is similar with that of zero-mean case. Now we can have the following expression:

$$\delta'(\sigma) = \begin{cases} \frac{-2k}{b}\left(\frac{1}{\sigma_f' - \sigma_m}\right)^{-1/b}(\mid \sigma - \sigma_m \mid)^{-(1+b)/b} & \text{for } \sigma > \sigma_m, \sigma \text{ increasing} \\ \frac{-2(1-k)}{b}\left(\frac{1}{\sigma_f' - \sigma_m}\right)^{-1/b}(\mid \sigma - \sigma_m \mid)^{-(1+b)/b} & \text{for } \sigma < \sigma_m, \sigma \text{ decreasing} \end{cases} \tag{2.43}$$

and for $\sigma < \sigma_m, \sigma$ decreasing

$$\dot{D}(t) = \frac{d\delta}{d\sigma}\frac{d\sigma}{dt} = -\frac{2}{b\sigma_f'}\left(\frac{\sigma}{\sigma_f'}\right)^{-(1+b)/b}\frac{d\sigma}{dt}. \tag{2.44}$$

2.4 Life prediction

Life prediction of a component is based on the past history of stress (or strain) on this component to predict the future life span of the component; the nature of this makes it difficult, if not impossible, to produce exact accurate result. Summarizing the methods in literature, we discuss three techniques used to estimate the remaining life period of the component monitored.

- *P-M Based Method*: Eq. (2.1) can be applied to life prediction for a repeating history by summing cycle ratios during only one repetition of the history, and then multiplying this by the unknown number of repetitions to failure (R):

$$\left[\sum_{\text{one rep.}} \left(\frac{n_i}{N_i}\right)\right] R = 1.$$

- *Non-Linear Based Method*: Considering the sequence effect, Eq. (2.2) instead of Eq. (2.1) should be used. Although non-linear methods may give better predictions than the P-M technique, the parameter P_i, depending on both the material type and shape, requires a considerable amount of testing.

- *Including Mean Stress Effect*: Assume that the strain-life and cyclic stress-stain curves for completely reversed cycling are represented by the following equations:

$$\begin{aligned} \epsilon_a &= \frac{\sigma_f'}{E}(2N)^b + \epsilon_f'(2N)^c, \\ \epsilon_a &= \frac{\sigma_a}{E} + \left(\frac{\sigma_a}{K'}\right)^{\frac{1}{n'}}. \end{aligned}$$

  To include the mean stress effect, Dowling [12] derived a formula by an analogous method to the modified Goodman diagram:

$$\epsilon_a = \frac{\sigma_f'}{E}(1 - \frac{\sigma_0}{\sigma_f'})(2N)^b + \epsilon_f'(1 - \frac{\sigma_0}{\sigma_f'})(2N)^c,$$

  ϵ_a here being the stress amplitude.

So integrating considerations of precision and implementation, we adopt the method of including mean stress effect based on P-M theory.

2.5 Comparison of two methods

It is difficult to evaluate a damage model because real data are needed. However, from the view of theory derivation, Ray's model is more accurate than that of Lorenzo's. Lorenzo's

Table 2.1: Nomenclature of some material constants

Symbol	Physical meaning	Value
s	cyclic strain hardening exponent	0.202
b	fatigue strength exponent	-0.108
c	fatigue ductility exponent	-0.500
A	strength coefficient (MPa)	1116
ϵ_f'	fatigue ductility coefficient	0.219
σ_f'	fatigue strength coefficient (MPa)	859

method is based on the two relations (Eq. (2.15) and Eq. (2.16)), but for most metal materials these two relations are just approximately correct. Now we can check for our case.

In our research, we use ASTM $A-516$ $Gr.$ 70 low carbon alloy steel. To make things clear, Eq. (2.15) and Eq. (2.16) are written here again and the nomenclature and value of the corresponding material constants can be found in Table 2.1.

$$\frac{1}{s} = \frac{c}{b}$$

and

$$A\epsilon_f'^{\,b/c} = A\epsilon_f'^{\,s} = \sigma_f'.$$

For Eq. (2.15), we have

$$\frac{1}{0.202} = 4.950 \simeq \frac{-0.500}{-0.108} = 4.630$$

and for Eq. (2.16)

$$1116 \times 0.219^{0.202} = 821.17 \simeq 859$$

From the above two expressions, we see that although acceptable, relations (2.15) and (2.16) do not exactly hold. But Lorenzo's model is simpler, thus more suitable for on-line control implementation.

Chapter 3

The Controlled Plant

3.1 Brief introduction to the controlled plant

3.1.1 Header system in Syncrude

LEC has found some applications including rocket engines and fossil power plants in a framework of feedforward and feedback control. In our research, a boiler-turbine system in an industrial co-generation system at Syncrude Canada's utility plant is the object of our study. The Syncrude Canada Ltd integrated energy facility is located in Mildred Lake, Alberta. Roughly 14% of Canada's domestic crude oil requirement is currently supplied by this plant, and there are plans to greatly increase this production in the near future [51]. Syncrude's Mildred Lake plant consists of a tarsand mine, a bitumen extraction process, and an upgrader to produce a high quality synthetic crude oil. Its Utility department is critical to plant operation because it supplies steam, water, electricity, air, and nitrogen to the various processes on the site. The main components of the steam system are three fuel gas fired utility boilers and two CO boilers which, with supplemental fuel gas firing, burn carbon monoxide and other waste gases from the cokers. These boilers supply a 900# header.

Syncrude Canada uses a complex header system for steam production and distribution, which includes headers at four different pressure levels (900,600,150 and 50 *psi*). The setup of the overall header system is shown in Fig. 3.1. The 900# header collects steam from three utility boilers and two CO boilers. The steam is then distributed for three different usages: i) to extract bitumen for oil sand; ii) to numerous turbines to generate electricity; iii) to three other headers to generate steam at different pressure. The overall plant, like many similar available world-wide, is thus a rather complex, nonlinear, interconnected system.

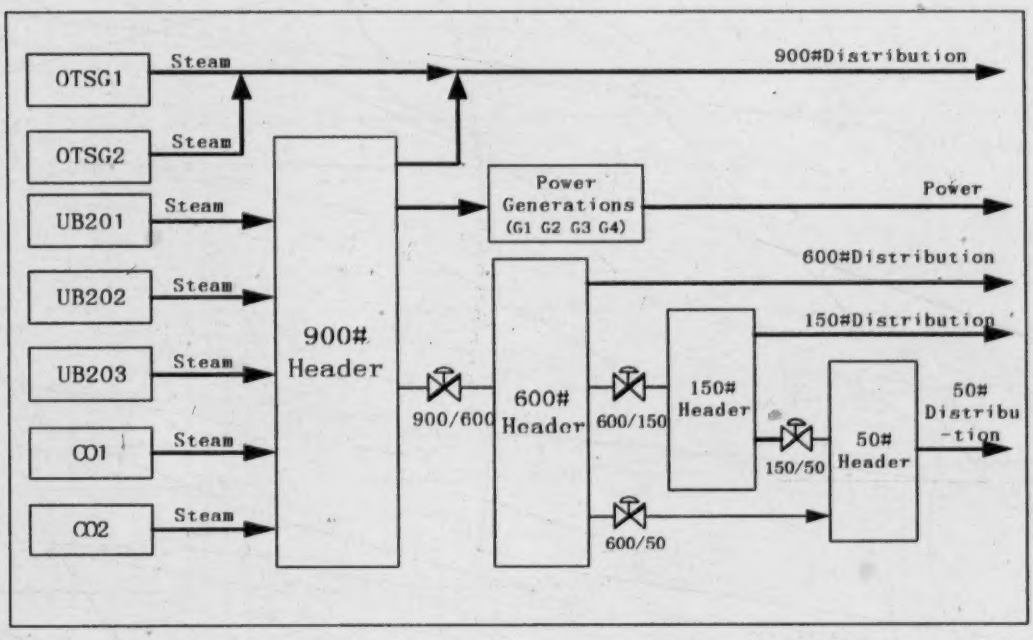

Figure 3.1: A schematic diagram of the steam header system with Syncrude

3.1.2 Utility Boiler

The operating principle of the utility boilers in the plant is shown in Fig. 3.2 [50]. The utility boilers in the plant are water tube drum boilers. This type of boiler usually comprises two separate subsystems: the steam water system (also called water side), and the fuel-air-fuel gas systems (also called the fire side).

In the steam-water side, water is preheated by economizer and then is fed into the steam drum. Liquid water exits the steam drum and flows through the downcomers into the mud drum. The mud drum distributes the water to the risers, where the water is heated then re-enters the steam drum in which the steam is separated from the water and exits the steam drum into the primary and secondary superheaters. In the two superheaters, the steam is further heated, and then is fed into the 900# header. In between the two superheaters is an attemperator which regulates the temperature of the steam exiting the secondary superheater by mixing water at a lower temperature with the steam from the primary superheater. In the fuel-air-flue side, fuel and air are thoroughly mixed and ignited in a furnace.

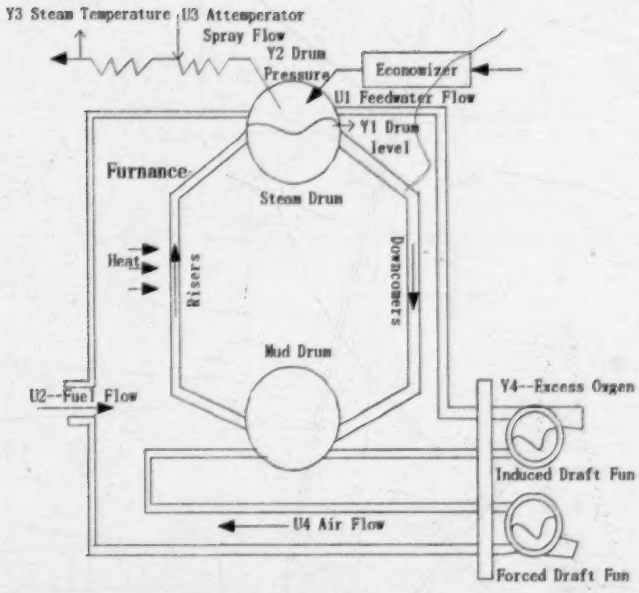

Figure 3.2: The operating principle diagram of the utility boiler

3.2 SYNSIM model

In the present case, Syncrude Canada Inc has available a simulation package, known as SYNSIM model [42]. SYNSIM is a computer simulation model that has been developed as a joint effort by members of Departments of Chemical Engineering and Electrical Engineering, University of Alberta, and of the technical staff of Syncrude Canada Ltd. The project was funded through a collaborative Research and Development grant from Syncrude Canada, with matching funds from the Natural Sciences and Engineering Research Council of Canada. The overall model includes submodels of the Utility and CO boilers with their controls, the 900#, 600#, 150# and 50# headers, and associated letdown system controls, the turbine-generator sets with their controls, and the plant electrical load with dynamic load-shedding. Work on this project began in May of 1994 and was completed by March 31, 1997. The purposes for which the model was developed included, but were not restricted to , the following:

1. Testing, through computer simulation, of the stability limits of the system under conditions of plant upset due to events such as tieline separation and loss of import power, tripping of one or more boilers and/or one or more turbines, stepwise changes in plant electrical or process steam loads, or any one or combination of many other

possible failures of plant equipments.

2. Testing of new, advanced control algorithms to be proposed, following the research of this project, with the goals of (1) improving the dynamic stability of this or similar co-generation plants under transient upset conditions, and (2) investigating the possibility of using pattern-recognition techniques to provide prediction/early detection of impending failure modes, interfacing this capability with the operating control system so as to reduce the frequency and/or severity of plant upsets.

3. Examination of the steam and electrical balances and stability limits with major changes or expansions of the utility system that may be proposed, in support of the planning process.

3.3 Linearized model

It is difficult to design a controller directly from a complicated nonlinear model, so a linearized model from SYNSIM model is first used and the resultant controller will also be simulated in SYNSIM. To simplify the control design problem, we assume the ratio of fuel-to-air flow rate be kept constant. With this assumption, the model becomes three imputes and three outputs. The definitions of u_i and y_i $(i = 1,\ 2,\ 3)$ can be found in Section (1.4).

The linearized plant model:

$$\begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix} = \begin{bmatrix} g_{11} & g_{12} & 0 \\ g_{21} & g_{22} & g_{23} \\ g_{31} & g_{32} & g_{33} \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix},$$

where

$$\begin{aligned}
g_{11} &= \frac{-0.00016s^2 + 0.000052s + 0.0000014}{s^2 + 0.0168s} \\
g_{12} &= \frac{0.0031s - 0.000032}{s^2 + 0.0215s} \\
g_{21} &= \frac{-0.0395}{s + 0.0018} \\
g_{22} &= \frac{0.312}{s + 0.0157} \\
g_{23} &= \frac{0.0853s^2 + 0.02924s + 0.000131}{s^2 + 0.03528s + 0.000142} \\
g_{31} &= \frac{-0.00118s + 0.000139}{s^2 + 0.01852s + 0.000091} \\
g_{32} &= \frac{0.448s + 0.0011}{s^2 + 0.0127s + 0.000095} \\
g_{33} &= \frac{0.582s - 0.0243}{s^2 + 0.1076s + 0.00104}
\end{aligned}$$

The operating conditions:

$$u_0 = \begin{bmatrix} u_{10} \\ u_{20} \\ u_{30} \end{bmatrix} = \begin{bmatrix} 63.84 \\ 3.399 \\ 0.1652 \end{bmatrix}, \quad y_0 = \begin{bmatrix} y_{10} \\ y_{20} \\ y_{30} \end{bmatrix} = \begin{bmatrix} 1.0 \\ 6306 \\ 500 \end{bmatrix}.$$

From the form of the plant model, we see that the first output, drum level, just depends on u_1 and u_2 and has nothing to do with u_3, and through simulation, we know only u_1 has a dominant effect on both y_1 and y_2, so it is difficult to design an LEC to satisfy all requirements on the three outputs at the same time directly. From the analysis above, a pre-compensator is needed. For the following LQR and MPC designs, the pre-compensator should make the plant stable and the resultant system is balanced in the sense of input-output. Fortunately, the original controller which is for the system performance can be used in this purpose. This controller, which is of PI type, is as follows:

The original controller K:

$$\begin{bmatrix} 206.94 & -0.0919 & -0.290 \\ 2.50 & 0.0055 & 0.0004 \\ 12.08 & 0.0099 & -0.0642 \end{bmatrix} + \begin{bmatrix} 2.20 & 0.0022 & -0.0021 \\ 0.027 & 0.0002 & 0 \\ 0.1355 & 0.0003 & -0.009 \end{bmatrix} \frac{1}{s}.$$

Chapter 4

Life Extending Controller Design with Linear Quadratic Regulation Theory

There are two main topics in LEC: fatigue damage modeling and life extending controller design. In Chapter 2, we discussed the issue of modeling. In this chapter and next chapter, we will concentrate on the design of a life extending controller. The main idea of LEC is to design a controller to get a better tradeoff between the durability of structure and system performance. Such an idea falls into the general framework of the linear quadratic (LQ) theory very well. So in this chapter we use LQ theory to design a state feedback LEC. Because not all states are available for measurement, an observer should be designed before the LEC can be implemented.

4.1 Problem formulation

To design an LEC via optimization, we formulate the LEC design problem as follows:

Given the plant dynamics:

$$\begin{aligned} \dot{x} &= Ax(t) + Bu(t), \quad x(t_0) = x_0, \\ y(t) &= Cx(t) + Du(t), \end{aligned} \tag{4.1}$$

where $y(t)$ and $u(t)$ are, respectively, the output and input vectors of the system. In our research, the output vector includes the drum level (m), drum pressure (psi) and steam temperature (^{0}C) and input vector includes feedwater flow (kg/sec), fuel flow (kg/sec) and attemperator spray flow (kg/sec).

The damage dynamics:

$$\frac{d\delta}{dt} = h(\delta(t), \sigma(t), t), \ \delta(t_0) = \delta_0, \tag{4.2}$$

where δ is a damage variable, the instantaneous damage, and σ is stress which is a function of the system outputs.

And the cost Function:

$$J = \int (Y^T Q Y + U^T R U + \dot{\delta}^T S \dot{\delta}) dt,$$

where Q, R and S are the weighting matrices for outputs, inputs, and damage variables, respectively. The task is to design a damage feedback controller to minimize J subject to Eq. (4.1) and Eq. (4.2).

In fact, two different cost functions are used to design controllers to show the effect of LEC. The first one is given by

$$J_1 = \int (q_1 y_1^2 + q_2 y_2^2 + q_3 y_3^2 + U^T R U) dt,$$

where y_1, y_2, y_3 are the plant outputs (thus J_1 does not account for the damage variables); the second is

$$J_2 = \int (q_1 y_1^2 + q_2 y_2^2 + q_3 y_3^2 + q_4 \dot{\delta}^2 + U^T R U) dt,$$

which includes the damage rate ($\dot{\delta}$), the output of damage model I. Because the damage dynamics represented by Eq. (4.2) is nonlinear, linearization should be performed first at the normal load conditions.

Because not all state variables are measurable, the state estimator is needed. The damage dynamics $v(t)$ is used to indicate the damage level at one or more critical points.

In Chapter 2, we have constructed damage model as the two following formulae:

$$\begin{aligned}
\frac{d\delta_e}{dt} &= 2 \times \frac{d}{d\sigma}\left(\left(\frac{\sigma - \sigma_r}{2(\sigma_f' - \sigma_m)}\right)^{-\frac{1}{b}}\right) \times \frac{d\sigma}{dt}, \\
\frac{d\delta_p}{dt} &= 2 \times \frac{d}{d\sigma}\left(\left(\frac{1}{\epsilon_f'} \times \left(\frac{\sigma - \sigma_r}{2K'}\right)^{\frac{1}{n'}}\right) \times \left(1 - \frac{\sigma_m}{\sigma_f'}\right)^{-\frac{c}{b}}\right)^{-\frac{1}{c}}\right) \times \frac{d\sigma}{dt}.
\end{aligned}$$

The physical meaning of all symbols can be found in Chapter 2. The material type of the turbine blade is known to be ASTM A-516 Gr.70. For damage modeling, the experiment constants in the above two equations can be found in appropriate tables:

$$\nu = 0.27, \quad E = 209.5 * 10^3, \quad \alpha = 0.00001,$$

$$K' = 1193 \text{ MPa}, \quad n' = 0.202, \quad \sigma_f' = 859,$$

$$\epsilon_f' = 0.219, \quad b = -0.108, \quad c = -0.5.$$

4.2 Design of an observer

As the discussion above, not all states are measurable, so a state observer has to be designed before LQ design. It is normal to design an observer using the pole arrangement method.

To check the effect of our observer, we develop a MATLAB function *Observer* to compare the system state change and the observer state change upon a step change of inputs. This function can be called by the following command:

$$[xh, x, t] = observer(sys, p, m);$$

where sys is the transfer function of the system; p is the desired poles set. This function can return states of the system x, states of the estimator xh and choose the time vector t by itself. Because this function is for MIMO system, the number of inputs m is also needed. Here we present the code of this program.

```
clearall;
sys = minreal(sys);
A = sys.a; B = sys.b; C = sys.c; D = sys.d;
H = place(A', C', p)';
[y, t, x] = step(sys);
G = ss(sys); A = G.a; B = G.b; C = G.c; D = G.d;
[y1, t1, xh] = step(ss((A - H * C), (B - H * D), C, D), t);
xh1 = 0;
yy = 0;
for i = 1 : m
xh1 = xh1 + xh(:, :, i);
yy = yy + y(:, :, i);
end
[y2, t2, xh2] = lsim(ss((A - H * C), H, C, D), yy, t);
xh = xh1 + xh2;
xx = 0;
for i = 1 : m
xx = xx + x(:, :, i);
end
figure(1)
plot(t, xx(:, 1 : 5), '-', t, xh(:, 1 : 5), ':')
figure(2)
plot(t, xx(:, 6 : 10), '-', t, xh(:, 6 : 10), ':')
figure(3)
plot(t, xx(:, 11 : 15), '-', t, xh(:, 11 : 15), ':')
figure(2)
plot(t, xx(:, 16 : 19), '-', t, xh(:, 16 : 19), ':')
```

Using the above function, we design an observer for the linearized model and the comparison of states of the system and observer upon a step input is shown in Fig. 4.1 to

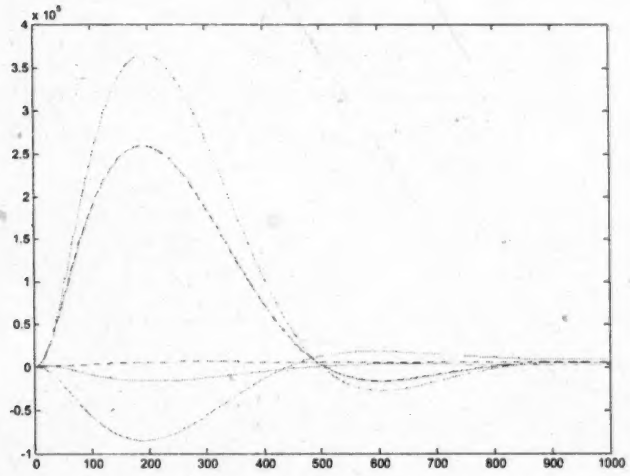

Figure 4.1: Comparison of states of the system and observer (states 1 to 5)

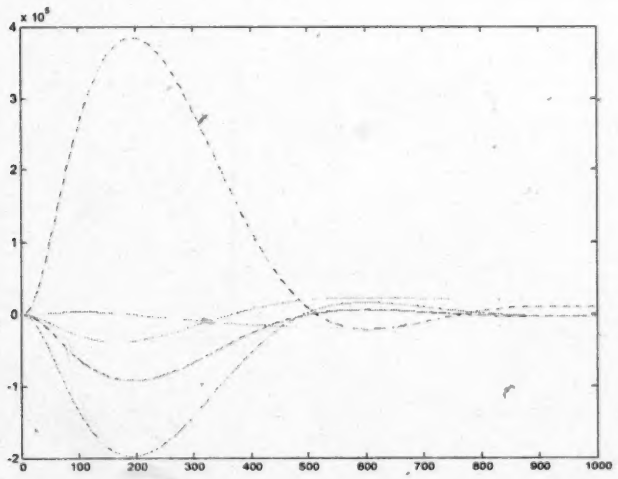

Figure 4.2: Comparison of states of the system and observer (states 6 to 10)

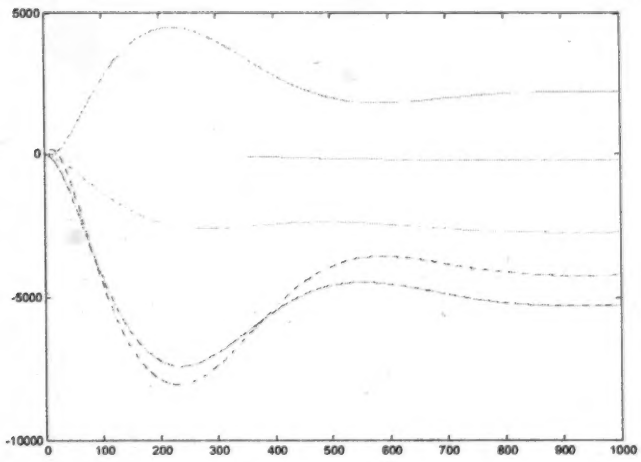

Figure 4.3: Comparison of states of the system and observer (states 11 to 15)

Fig. 4.4. In all figures, solid lines represent the "true states" while dotted lines show the estimated states. We see that the states of the estimator match the states of the system very well; they are not distinguishable from the graphs.

4.3 Linear quadratic optimal algorithm

Before linear quadratic regulator (LQR) design, the state-feedback method for assigning the closed-loop eigenvalues to a certain specified region in the left s-plane had found wide applications, but the resultant state-feedback gain is not unique. It means that other performance can be considered in addition to stability. Meanwhile because there always exist some constraints on the magnitude of inputs of a system, a trade-off between the performance and the magnitude of inputs is needed. To consider the trade-off and minimize the cost function, a criterion that judges "optimal" or "best" performance is the main idea of LQ. Algebraic Riccati equation (ARE) algorithm is one of most commonly used LQ methods; it gives out the optimal solution by solving some algebraic Riccati equation.

ALGEBRAIC METHOD [7, 56] Given A, P, Q as real $n \times n$ matrices, P and Q symmetric, define a $2n \times 2n$ matrix

$$H \triangleq \begin{bmatrix} A & -P \\ -Q & -A^T \end{bmatrix}.$$

This is called a Hamiltonian matrix. It is easy to show that eigenvalues of H are symmetric

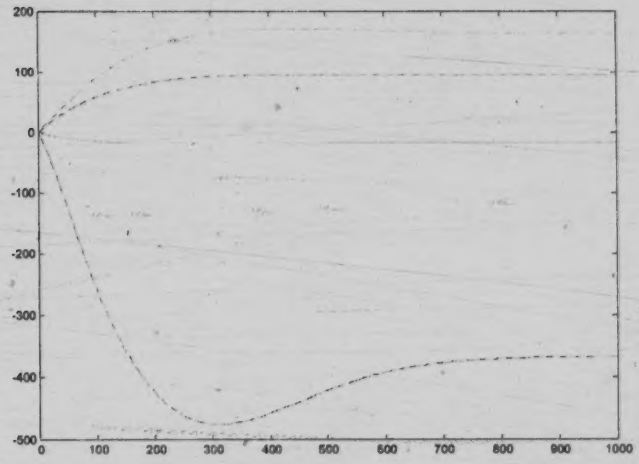

Figure 4.4: Comparison of states of the system and observer (states 16 to 19)

about the imaginary axis in the s-plane, i.e; H and $-H^T$ are similar, thus if λ is an eigenvalue of H, then $-\lambda$ is also an eigenvalue of H. Now we have the following conclusion.

conclusion 1 *Assume H has no eigenvalue on the imaginary axis. Then H must have n eigenvalues in the left half s-plane and n eigenvalues in the right half s-plane, denoted as $\{\lambda_i^- : i = 1, 2, ...n\}$, $\{\lambda_i^+ : i = 1, 2, ...n\}$ respectively.*

The corresponding eigenvectors are also separated into two subspaces $U^-(H)$ and $U^+(H)$. Now we focus on $U^-(H)$ and construct it as

$$U^-(H) = \mathrm{Im}\begin{bmatrix} X_1 \\ X_2 \end{bmatrix},$$

where $X_1,\ X_2 \in C^{n\times n}$. If $U^-(H)$ can be arranged in a way that X_1 is non-singular, then we can define $X = X_2 * X_1^{-1}$. So X is uniquely determined by H and it has the following properties:

1. X is symmetric;
2. X satisfies the equation:

$$A^T X + XA - XPX + Q = 0;$$

3. $A - PX$ is stable.

Now we can prove that by choosing $u^* = -KX$ with $K = R^{-1}B^TX$, and X being the unique positive definite solution of

$$A^TX + XA - XBR^{-1}B^TX + Q = 0,$$

u^* is the unique optimal solution. For this u^*, $J^* = x_0^TXx_0$. From the discussion above, we can get a design procedure of the LQR:

1. solving the ARE (numerically)

$$A^TX + XA - XBR^{-1}B^TX + Q = 0;$$

2. calculate $K = R^{-1}B^TX$;

3. $(A - BK)$ is stable, and J^* is minimum with $u^* = -Kx$, $J^* = x_0^TXx_0$.

For the reason of saving space, all proofs for the conclusions in this section are omitted. They can be found in references [7, 56]. According to the above procedure, we can design our life extending controller. To show the effect of LEC, two different cost functions J_1 and J_2 are used. Controllers 1 and 2 are designed by minimizing J_1 and J_2, respectively, setting all weightings to unity. The controllers are of high-order (with 19 states). Because of the way cost functions are defined, controller 2 is an LEC; while controller 1 is not.

4.4 Simulation results

Simulations are conducted using the linearized boiler model first under normal load conditions, the results are shown in Figs. 4.9 to 4.12, where solid lines represent responses with controller 2 (with LEC) and dashed lines represent those with controller 1 (without LEC).

To test the validity of the LEC, simulations using the nonlinear model SYNSIM of the boiler system are also presented. The results are shown in Fig. 4.9 to Fig. 4.12, where solid lines represent responses with controller 2 (with LEC) and dashed lines represent those with controller 1 (without LEC). It is clear that the LEC achieves its objective of significantly reducing the accumulated damage in the system on both the linear and nonlinear models.

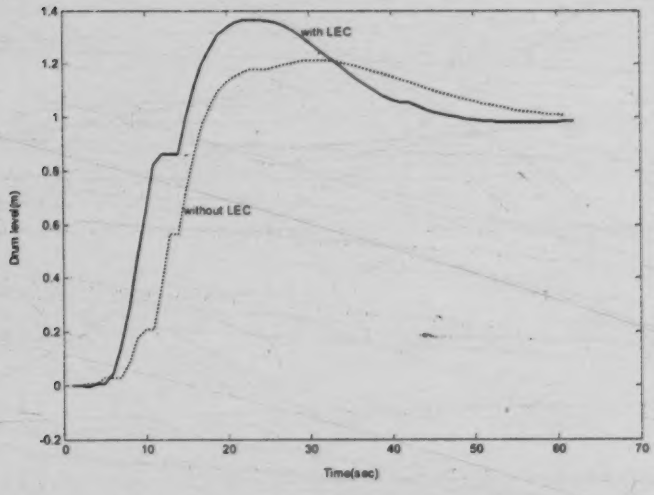

Figure 4.5: Step response of the drum level with the linear model

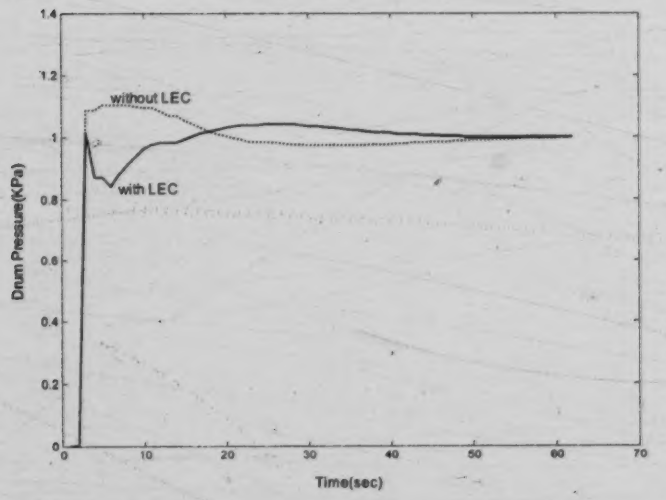

Figure 4.6: Step response of the drum pressure with the linear model

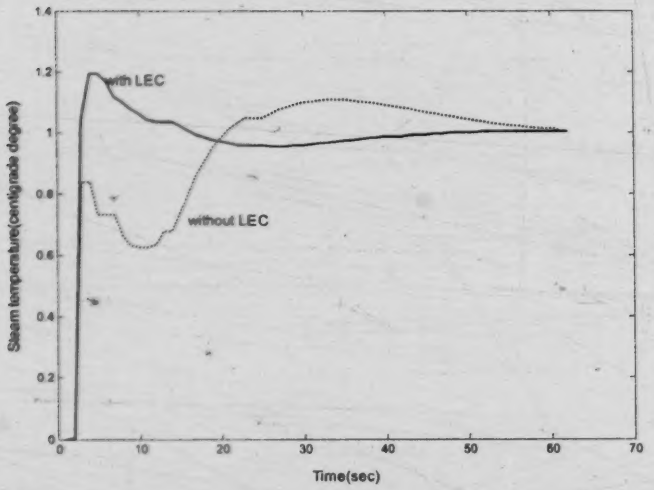

Figure 4.7: Step response of the steam temperature with the linear model

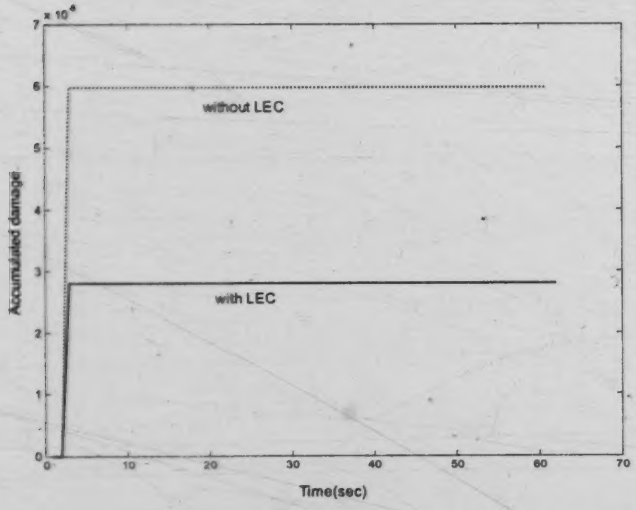

Figure 4.8: Comparison of the accumulated damage with the linear model

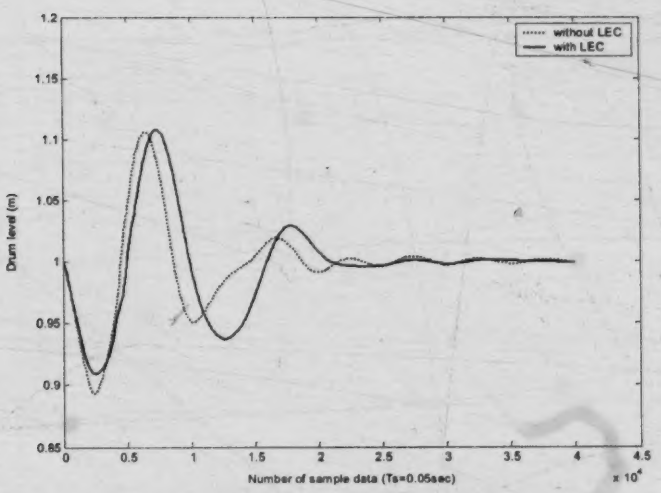

Figure 4.9: Step response of the drum level with the nonlinear model

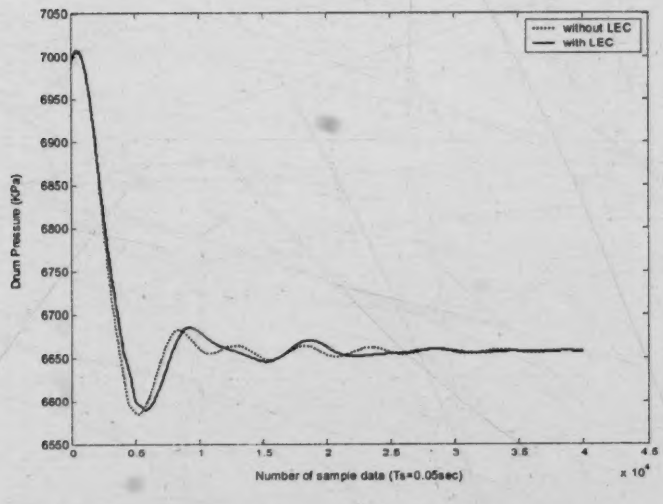

Figure 4.10: Step response of the drum pressure with the nonlinear model

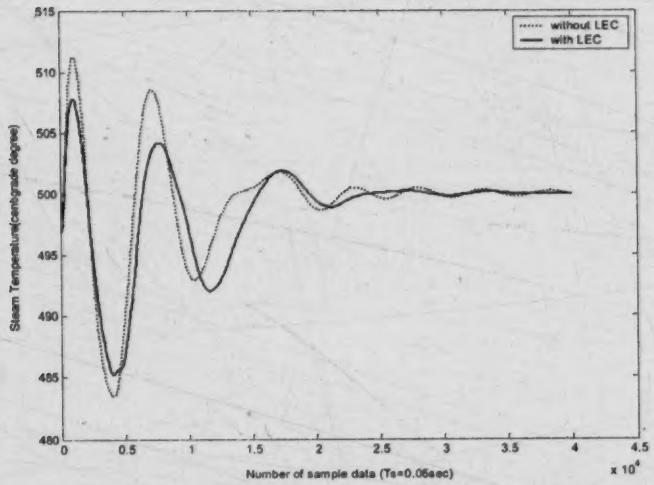

Figure 4.11: Step response of the steam temperature with the nonlinear model

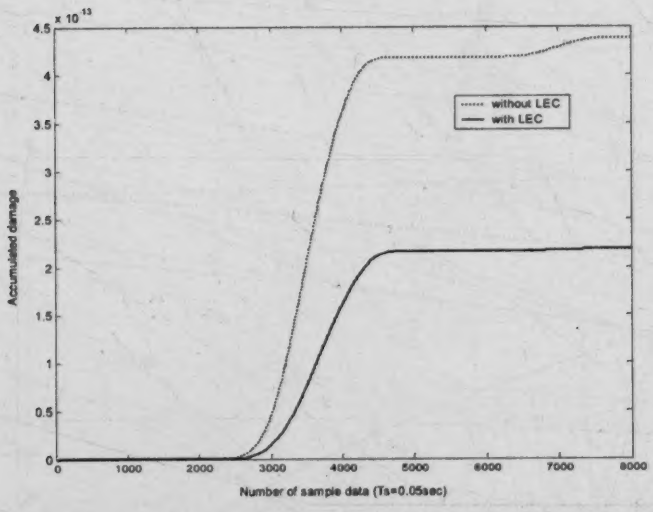

Figure 4.12: Comparison of the accumulated damage with the nonlinear model

Chapter 5

Life Extending Controller Design with Model Predictive Control Theory

5.1 The setup of our control system

We still use the linear model which is linearized from SYNSIM model [42] to design an LEC, and simulate using both the linear and SYNSIM models. The linear model is as follows:

$$\begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix} = \begin{bmatrix} g_{11} & g_{12} & 0 \\ g_{21} & g_{22} & g_{23} \\ g_{31} & g_{32} & g_{33} \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix},$$

where

$$\begin{aligned}
g_{11} &= \frac{-0.00016s^2 + 0.000052s + 0.0000014}{s^2 + 0.0168s} \\
g_{12} &= \frac{0.0031s - 0.000032}{s^2 + 0.0215s} \\
g_{21} &= \frac{-0.0395}{s + 0.0018} \\
g_{22} &= \frac{0.312}{s + 0.0157} \\
g_{23} &= \frac{0.0853s^2 + 0.02924s + 0.000131}{s^2 + 0.03528s + 0.000142} \\
g_{31} &= \frac{-0.00118s + 0.000139}{s^2 + 0.01852s + 0.000091} \\
g_{32} &= \frac{0.448s + 0.0011}{s^2 + 0.0127s + 0.000095} \\
g_{33} &= \frac{0.582s - 0.0243}{s^2 + 0.1076s + 0.00104}
\end{aligned}$$

There are constraints on three inputs: $0 \leq u_1 \leq 120$, $0 \leq u_2 \leq 7$ and $0 \leq u_3 \leq 10$. The original controller which is for the system performance is used as a pre-compensator. It is a PI type controller:

$$\begin{bmatrix} 206.94 & -0.0919 & -0.290 \\ 2.50 & 0.0055 & 0.0004 \\ 12.08 & 0.0099 & -0.0642 \end{bmatrix} + \begin{bmatrix} 2.20 & 0.0022 & -0.0021 \\ 0.027 & 0.0002 & 0 \\ 0.1355 & 0.0003 & -0.009 \end{bmatrix} \frac{1}{s}.$$

5.2 The reduction of the model

After the pre-compensation, the system becomes very complex, being 19th order. So it is necessary to reduce this system before further controller design. For this purpose, we adopt a system identification approach [24, 46]. Because it is a linear model, we can reduce the nine elements of the plant separately.

If a system can be represented by

$$y(t) = G(q^{-1})u(t) + H(q^{-1})e(t), \tag{5.1}$$

where $G(q^{-1})$ and $H(q^{-1})$ are models of plant and disturbance respectively, and control input $u(t)$ and noise signal $e(t)$ are independent, then there exist the following relationships among the input, output and cross spectral densities:

$$\phi_y(\omega) = \mid G(e^{j\omega}) \mid^2 \phi_u(\omega) + \mid H(e^{j\omega}) \mid^2 \lambda^2, \tag{5.2}$$

$$\phi_{yu}(\omega) = G(e^{j\omega})\phi_u(\omega), \tag{5.3}$$

where λ^2 is the variance of $e(t)$. The prediction error $\epsilon(t)$ is given by

$$\epsilon(t) = \frac{1}{\hat{H}}\left[(G - \hat{G})u(t) + He(t)\right]. \tag{5.4}$$

Recall that the prediction error method is to minimize

$$\hat{\theta} = arg_\theta \min \sum_{t=1}^{N} (\epsilon(t))^2 \stackrel{or}{=} arg_\theta \min \frac{1}{N} \sum_{t=1}^{N} (\epsilon(t))^2. \tag{5.5}$$

As the number of data points $N \to \infty$, we have

$$\hat{\theta} = arg_\theta \min E|(\epsilon(t))|^2. \tag{5.6}$$

Now using Parseval's Theorem, we have

$$\hat{\theta} \stackrel{N\to\infty}{\longrightarrow} arg_\theta \min \int_{-\pi}^{\pi} \left[\mid G(e^{j\omega}) - \hat{G}(e^{j\omega}) \mid^2 \frac{\Phi_u(\omega)}{\mid \hat{H}(e^{j\omega}) \mid^2} + \frac{\mid H(e^{j\omega}) \mid^2}{\mid \hat{H}(e^{j\omega}) \mid^2} \lambda^2 \right] d\omega. \tag{5.7}$$

What is the lower bound of this optimization problem? From

$$\epsilon(t) = \frac{1}{\hat{H}}\left[(G - \hat{G})u(t) + He(t)\right] \tag{5.8}$$

and by the constraint that $H(0) = \hat{H}(0) = 1$, we have

$$\epsilon(t) = e(t) + \text{terms independent of } e(t). \tag{5.9}$$

Therefore we have

$$E(\epsilon(t))^2 \geq E\,(e(t))^2 = \lambda^2.$$

Under what conditions is this lower bound achieved? The answer is $\hat{G} = G$ and $\hat{H} = H$, i.e. both plant and disturbance models converge to true dynamics. We choose output error (OE) model; setting $\hat{H} = \bar{H} = 1$, then we have

$$\hat{\theta} \overset{N\to\infty}{\longrightarrow} arg_\theta \min \int_{-\pi}^{\pi} \left[\mid G(e^{j\omega}) - \hat{G}(e^{j\omega}) \mid^2 \frac{\Phi_u(\omega)}{\mid \bar{H}(e^{j\omega}) \mid^2} + \frac{\mid H(e^{j\omega}) \mid^2}{\mid \bar{H}(e^{j\omega}) \mid^2} \lambda^2 \right] d\omega. \tag{5.10}$$

Since $\bar{H}$ is a constant, this optimization problem is equivalent to

$$\hat{\theta} \overset{N\to\infty}{\longrightarrow} arg_\theta \min \int_{-\pi}^{\pi} \left[\mid G(e^{j\omega}) - \hat{G}(e^{j\omega}) \mid^2 \frac{\Phi_u(\omega)}{\mid \bar{H}(e^{j\omega}) \mid^2} \right] d\omega. \tag{5.11}$$

Now the lower bound for this optimization problem becomes zero and it is easy to know this lower bound is achieved when $\hat{G} = G$. Therefore, the OE method gives a consistent estimate for a process.

There are several statistical tests on the prediction errors $\epsilon(t, \hat{\theta}_N)$. The prediction errors evaluated at the parameter estimate $\hat{\theta}_N$ are often called the residuals. To simply the notation, we will denote $\epsilon(t, \hat{\theta}_N)$ simply by $\epsilon(t)$.

To check the rightness of the estimated model, several tests can be constructed:

1. $\epsilon(t)$ is zero mean white noise;

2. $\epsilon(t)$ has a symmetric distribution;

3. $\epsilon(t)$ is independent of all inputs ($E\epsilon(t)u(s)) = 0$, for all t and s).

Consider the first test. If $\epsilon(t)$ is a white noise, then its autocovariance function $\hat{\gamma}(\tau)$ is zero except at $\tau = 0$. In practice, a normalized autocovariance function, also known as autocorrelation function $\rho_\epsilon(\tau)$, is often used:

$$\rho_\epsilon(\tau) = \frac{\hat{\gamma}_\epsilon(\tau)}{\hat{\gamma}_\epsilon(0)}.$$

To apply the normal test under the risk of 1%, a threshold can be chosen as 3. Then

1. If more than 1% $\rho_\epsilon(\tau)$ satisfy $\mid \rho_\epsilon(\tau) \mid > 3/\sqrt{N}$ where $\tau \neq 0$, then the model is invalidated;

2. If less than 1% $\rho_\epsilon(\tau)$ satisfy $\mid \rho_\epsilon(\tau) \mid > 3/\sqrt{N}$ where $\tau \neq 0$, then the model passes this test.

To perform the second test, a simple histogram may be used to test the symmetry of the residuals. To perform the third test, one can check the cross covariance between $\epsilon(t)$ and $u(t)$ to see if they are zero.

$$\hat{\gamma}_{\epsilon u}(\tau) = \frac{1}{N} \sum_{t=1-min(0,\tau)}^{N-max(\tau,0)} \epsilon(t+\tau)u(t). \tag{5.12}$$

In practice, the normalized cross covariance, also known as cross correlation $\rho_{\epsilon u}(\tau)$ is often used:

$$\rho_{\epsilon u}(\tau) = \frac{\hat{\gamma}_{\epsilon u}(\tau)}{\sqrt{\hat{\gamma}_\epsilon(0)\hat{\gamma}_u(0)}}. \tag{5.13}$$

Asymptotically, we can obtain

$$\sqrt{N}\rho_{\epsilon u}(\tau) \sim N(0,1).$$

To apply the normal test under the risk of 1%, a threshold can be chosen as 3. Then

1. If more than 1% $\rho_{\epsilon u}(\tau)$ satisfy $\mid \rho_{\epsilon u}(\tau) \mid > 3/\sqrt{N}$, then the model is invalidated.

2. If less than 1% $\rho_{\epsilon u}(\tau)$ satisfy $\mid \rho_{\epsilon u}(\tau) \mid > 3/\sqrt{N}$, then the model passes this test.

In MATLAB, the function *residue* provides both the whiteness test and the cross correlation test.

According to the discussion above, procedures to identify and test are as follows:

Step1. Step response test. This step is to understand the properties of the system and prepare for the design of a random binary input sequence (RBS).

Step2. RBS input design. According to the step response of the system, we can coarsely calculate its time delay and time constant, and then design a RBS input according to the corresponding experience formula.

Step3. Data pre-processing. It includes data segmentation, data detrend, outlier detection.

Step4. Parameter estimation and model test. In this step, the structure and order of models are determined, and the corresponding parameters are identified. Since the integrated GUI software 'ident' is the main tool in this step, the residual test and model validation are completed at the same time.

5.2.1 The identification of g_{11}, g_{21} and g_{31}

Step 1 Step response test: u_1−step change, $u_2 = 0$, $u_3 = 0$.

Fig. 5.1 to Fig. 5.3 show all three inputs and Fig. 5.4 to Fig. 5.6 all three outputs.

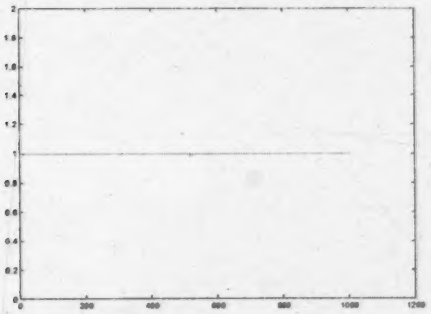

Figure 5.1: Input u_1

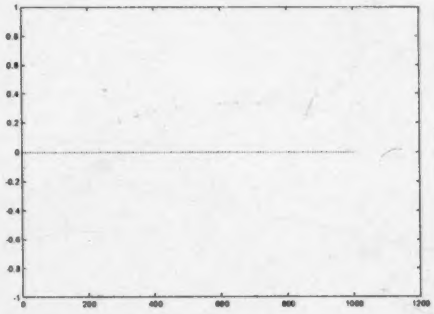

Figure 5.2: Input u_2

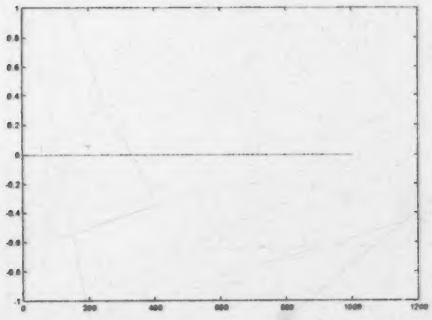

Figure 5.3: Input u_3

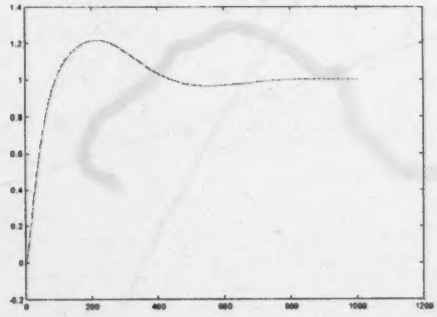

Figure 5.4: Output y_1: Drum level

Step 2 RBS input design.

Now according to the figure of y_1, we can calculate its time constants:

$$\tau_d = 0,\ \sigma_p = \frac{1.2-1}{1} = 20\%,\ t_p = 200s,$$

$$and, \sigma_p = exp(-\frac{\zeta\pi}{\sqrt{1-\zeta^2}}),\ \ln\sigma_p = -\frac{\zeta\pi}{\sqrt{1-\zeta^2}},$$

$$so, (\ln\sigma_p)^2 = \frac{\zeta^2\pi^2}{1-\zeta^2},\ t_p = \frac{\pi\tau}{\sqrt{1-\zeta^2}}$$

$$then, \zeta = 0.4559,\ \tau = 56.66.$$

Taking $T_s = 6s$, thus $f = \frac{k*T_s}{\tau\pi} = 0.1$. The following code can be used to produce the RBS input:

$clear\ all;$

$Ts = 6;$

$U = idinput(1000,'rbs',[0\ 0.1],[-50\ 50]);$

$U = [zeros(600,1);U];$

$T = [0 : Ts : Ts * (length(U) - 1)];$

$U = [T'\ U];$

Step 3 Data pre-processing: Segmentation and detrend.

The first reference input signal and the first output signal drum level are shown in Fig. 5.7. The first input reference signal and the second output signal drum pressure are shown in Fig. 5.9. The first reference input signal and the third output signal steam temperature are shown in Fig. 5.11. We use data (601-1300) for identification and data (1301-1600) for validation. The input and output signals after segmentation and detrend are shown in Fig. 5.8 , Fig. 5.10 and Fig. 5.12.

Step 4 Parameter estimation and model test.

Fig. 5.13 shows the output fit situation of the reduced model and the original of g_{11}. From it, we know the fit rate attains 97.58%. In Fig. 5.14, we check if the reduced model is correct by residual test. From the bottom part of Fig. 5.14, we know the process model g_{11} is correct holding a 99% confidence interval. The top part of Fig. 5.14 just shows the noise model is not good. This is mainly because of the fact we choose an OE model to identify; it doesn't affect the rightness of the reduced process model. Fig. 5.15 and Fig. 5.16 show the

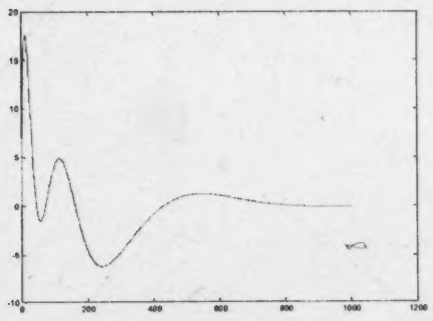

Figure 5.5: Output y_2: Drum pressure

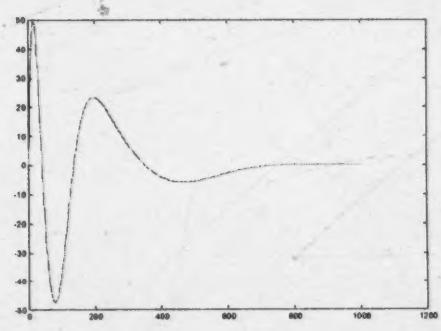

Figure 5.6: Output y_3: Steam temperature

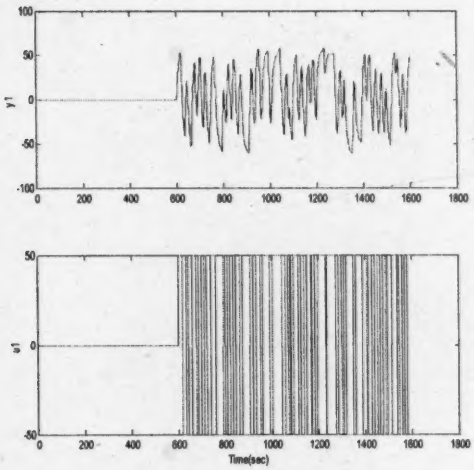

Figure 5.7: The first reference input and drum level output

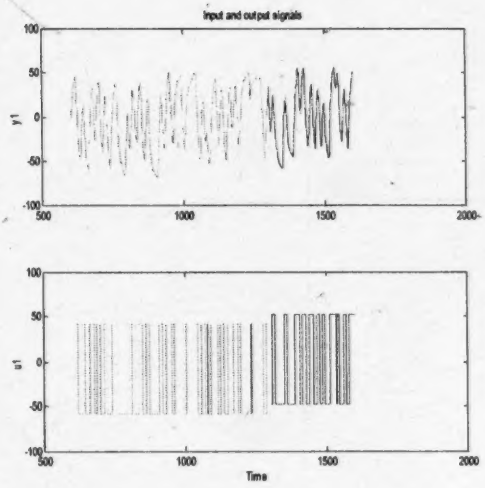

Figure 5.8: After segmentation and detrend of the input and output

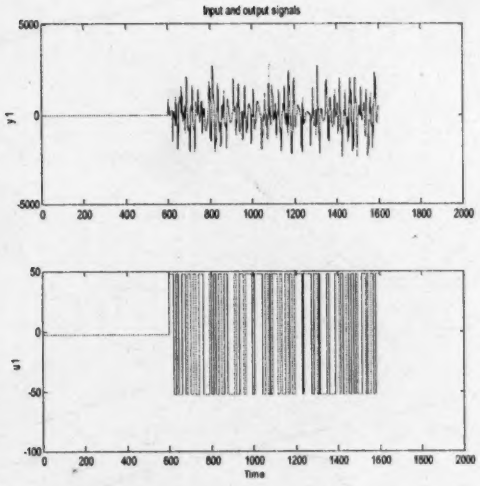

Figure 5.9: The first reference input and drum pressure output

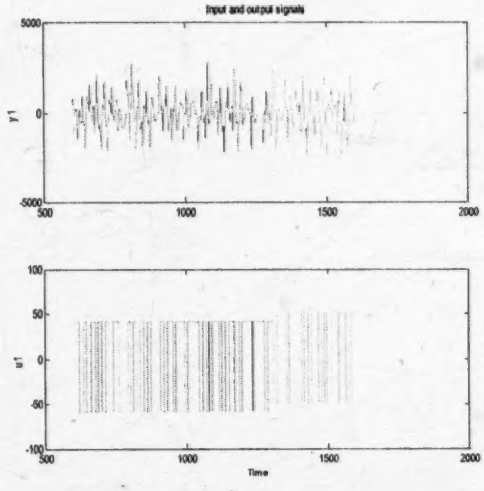

Figure 5.10: After segmentation and detrend of the input and output

fit rate and residual test for g_{21}, while Fig. 5.17 and Fig. 5.18 are for g_{31}. We can conclude from these figures all three process models are good enough.

Step5: Parameter acquirement and the model transfer from discrete to continuous.

g_{11}

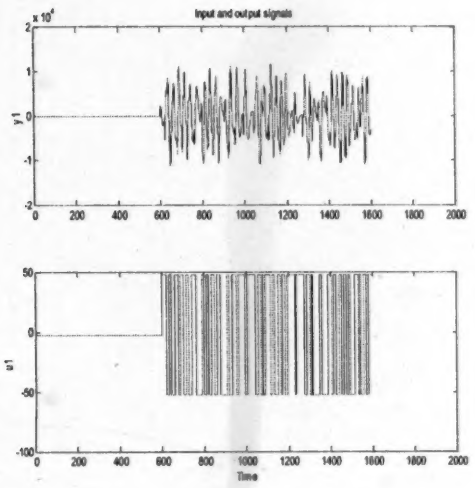

Figure 5.11: The first reference input and steam temperature

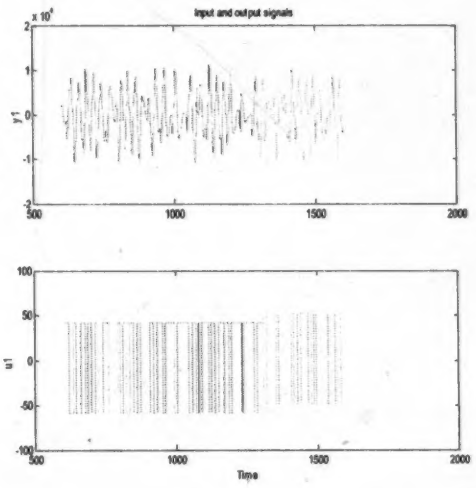

Figure 5.12: After segmentation and detrend of the input and output

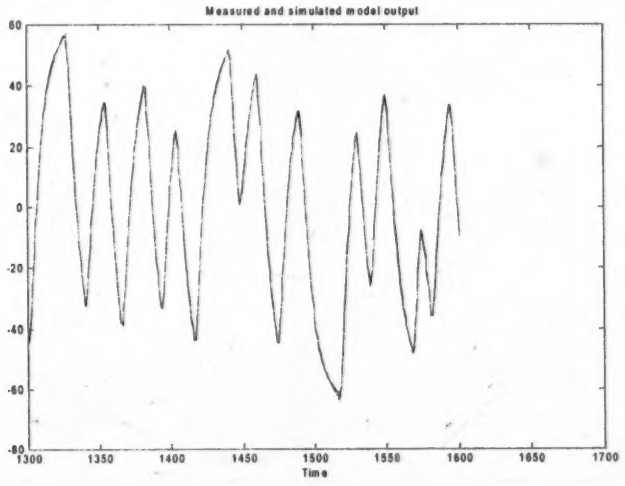

Figure 5.13: The output fit of g_{11}

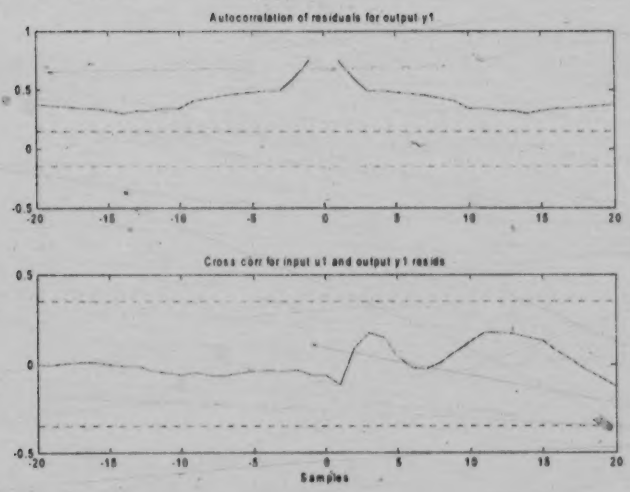

Figure 5.14: The residual test for g_{11}

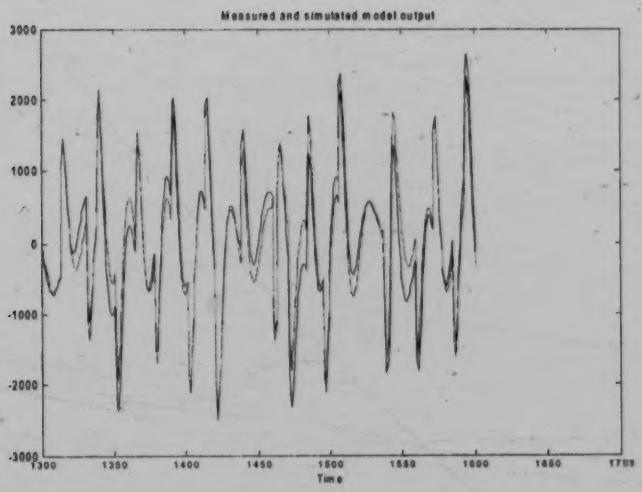

Figure 5.15: The output fit of g_{21}

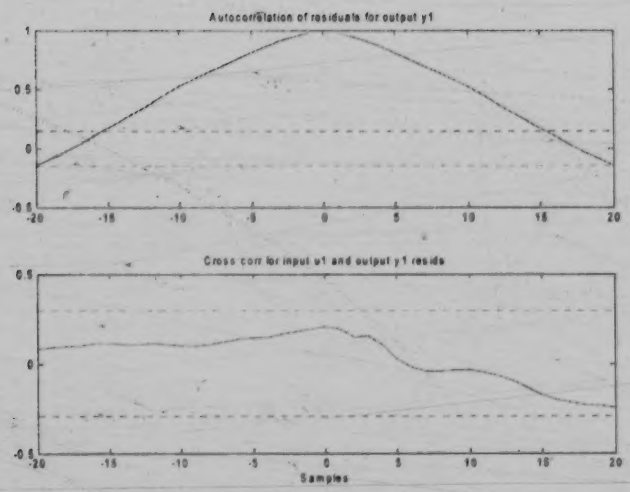

Figure 5.16: The residual test for g_{21}

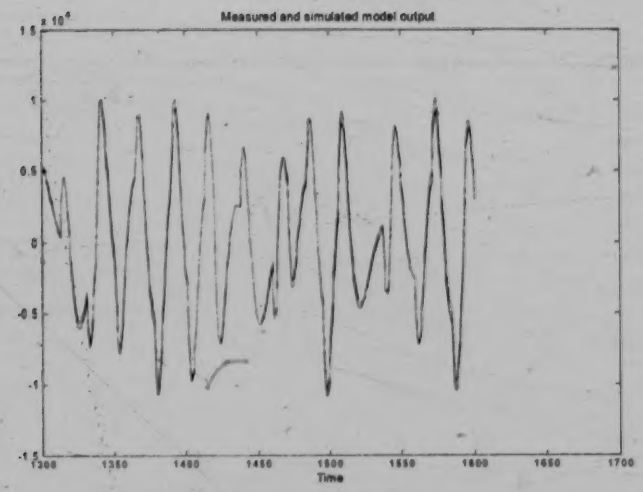

Figure 5.17: The output fit of g_{31}

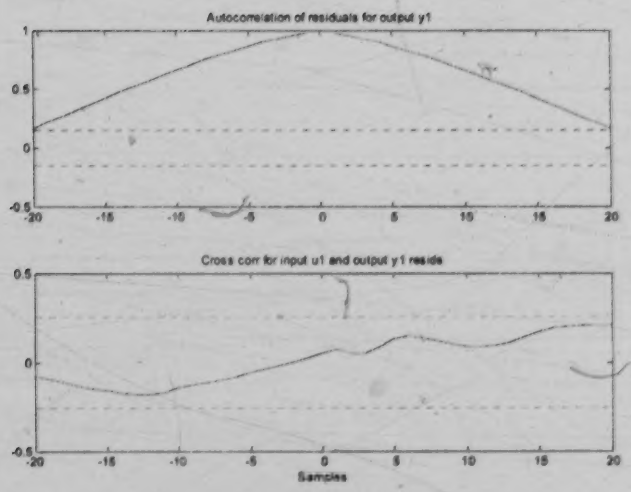

Figure 5.18: The residual test of g_{31}

Discrete parameters:

Discrete-time IDPOLY model: $y(t) = [B(q)/F(q)]u(t) + e(t)$

$B(q) = -0.00843(+-0.0009285) + 0.07463(+-0.003115)q^{-1}$

$\quad$ -0.119$(+-0.003504)q^{-2} + 0.05307(+-0.001325)q^{-3}$

$F(q) = 1 - 3.483(+-0.004199)q^{-1} + 4.547(+-0.01205)q^{-2}$

Now using the MATLAB function *d2c*, we can transfer the discrete time model to a continuous time model.

Continuous-time IDPOLY model: $y(t) = [B(s)/F(s)]u(t) + e(t)$

$B(s) = -0.00843s^4 + 0.01804s^3 + 0.06462s^2 + 0.006336\ s + 0.0003298$

$F(s) = s^4 + 0.5541s^3 + 0.1163s^2 + 0.006593\ s + 0.0003269$

g_{21}

Discrete parameters

Discrete-time IDPOLY model: $y(t) = [B(q)/F(q)]u(t) + e(t)$

$B(q) = -0.1917(+-0.4002) + 14.8(+-0.8461)q^{-}1$

$\quad -27.96(+-0.7441)q^{-}2 + 13.65(+-0.4283)q^{-}3$

$F(q) = 1 - 2.177(+-0.01466)q^{-}1 + 1.696(+-0.03385)q^{-}2$

$\quad$ - 0.5659 (+-0.05296) $q^{-}3 + 0.1002(+-0.02706)q^{-}4$

Continuous parameter

Continuous-time IDPOLY model: $y(t) = [B(s)/F(s)]u(t) + e(t)$

$B(s) = -0.1917s^4 + 16.17s^3 + 44.25s^2 + 2.912s + 0.9307$

$F(s) = s^4 + 2.3s^3 + 2.427s^2 + 0.5504s + 0.164$

g_{31}

Discrete-time IDPOLY model: $y(t) = [B(q)/F(q)]u(t) + e(t)$

$B(q) = 0.1367(+-0.6574) + 26.99(+-1.898)q^{-}1$

$\quad$ - 57.49 (+-2.029) q^{-}2 + $30.44(+-0.8022)q^{-}3$

$F(q) = 1 - 2.766(+-0.01618)q^{-}1 + 2.797(+-0.04849)q^{-}2$

$\quad -1.213(+-0.04928)q^{-}3 + 0.1891(+-0.01715)q^{-}4$

Continuous-time IDPOLY model: $y(t) = [B(s)/F(s)]u(t) + e(t)$

$B(s) = 0.1367s^4 + 24.91s^3 + 61.42s^2 - 6.874s + 0.1811$

$F(s) = s^4 + 1.665s^3 + 0.8306s^2 + 0.1486s + 0.01757$

5.2.2 The identification of g_{12}, g_{22} and g_{32}

Step 1 Step response test: u_2–step change, $u_1 = 0$, $u_3 = 0$

Fig. 5.19 to Fig. 5.21 show all three inputs and Fig. 5.22 to Fig. 5.24 all three outputs.

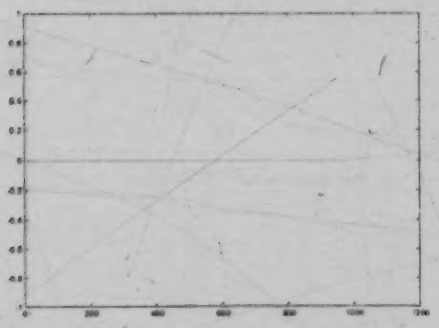

Figure 5.19: Input u_1

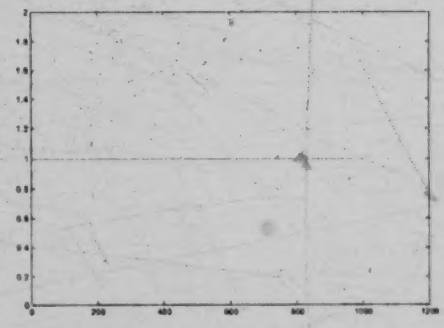

Figure 5.20: Input u_2

Step 2 RBS input design.

Now according to the figure of y_2, we can calculate its time constant: $\tau = 30(sec)$.

So, we take $T_s = 3s$, thus $f = \frac{k_s T_s}{\tau \pi} = 0.1$. The following code can be used to produce the RBS input:

clear all;

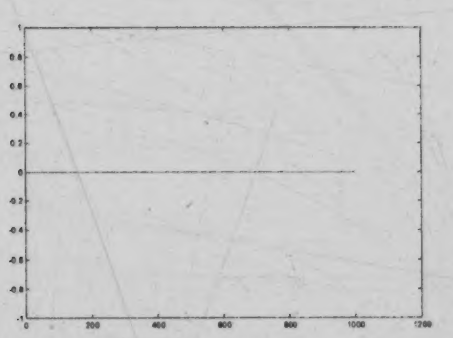

Figure 5.21: Input u_3

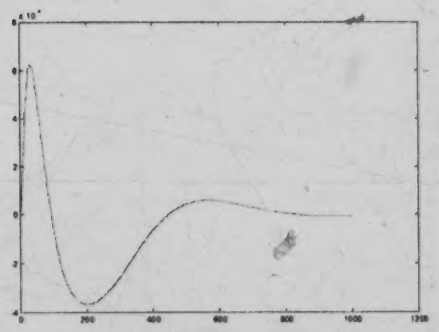

Figure 5.22: Output y_1: Drum level

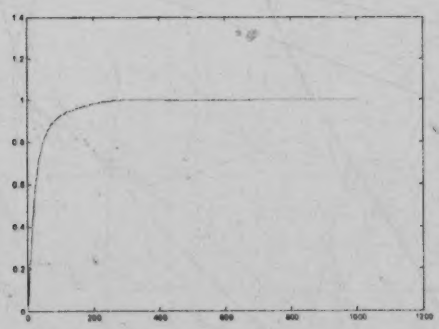

Figure 5.23: Output y_2: Drum pressure

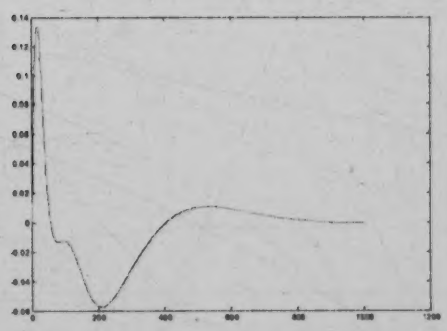

Figure 5.24: Output y_3: Steam temperature

$Ts = 3;$

$U = idinput(1000,'rbs',[0\ 0.1],[-50\ 50]);$

$U = [zeros(600,1);U];$

$T = [0 : Ts : Ts * (length(U) - 1)];$

$U = [T'\ U];$

Step 3 Data pre-processing: Segmentation and detrend.

The second reference input signal and the first output signal drum level are shown in Fig. 5.25. The second reference input signal and the second output signal drum pressure are shown in Fig. 5.27. The second reference input signal and the third output signal steam temperature are shown in Fig. 5.29. We use data (601-1300) for identification and data (1301-1600) for validation. The input and output signals after segmentation and detrend are shown in Fig. 5.26 , Fig. 5.28 and Fig. 5.30.

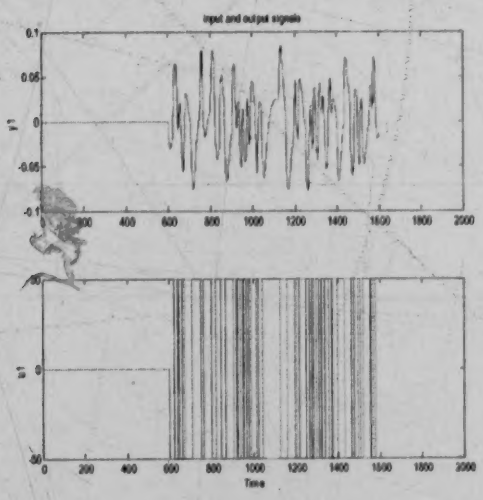

Figure 5.25: The second reference input and drum level output

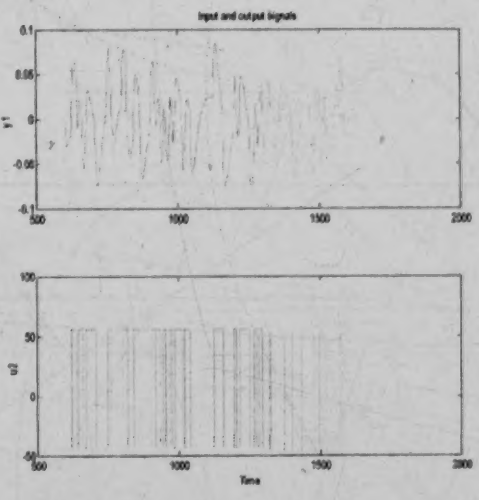

Figure 5.26: After segmentation and detrend of the input and output

Step 4 Parameter estimation and model test.

Fig. 5.31 shows the output fit situation of the reduced model and the original of g_{12}. From it, we know the fit rate attains 78%. In Fig. 5.32, we check if the reduced model is correct by residual testa. From the bottom part of Fig. 5.32, we know the process model g_{12} is correct holding a 99% confidence interval. The top part of Fig. 5.32 just shows the noise model is not good. This is mainly because of the fact we choose an OE model to identify; it

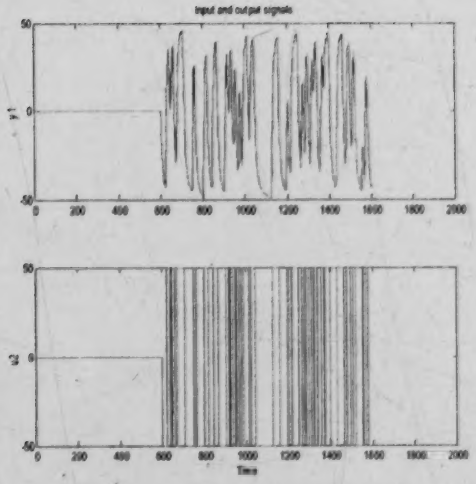

Figure 5.27: The second reference input and drum pressure output

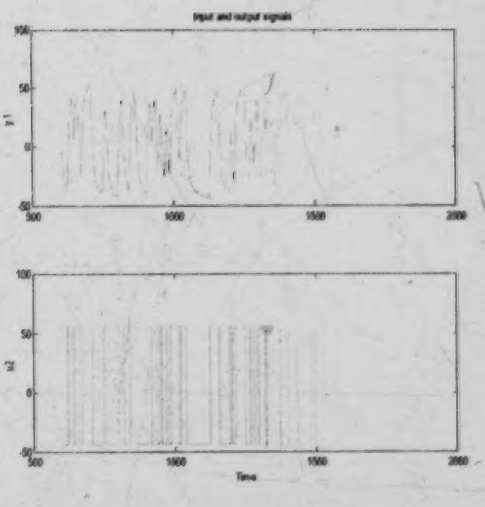

Figure 5.28: After segmentation and detrend of the input and output

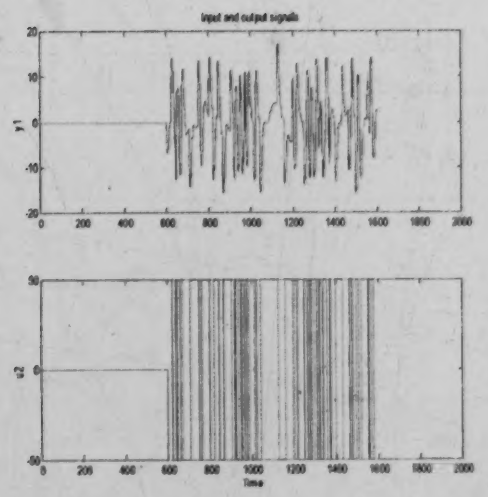

Figure 5.29: The second reference input and steam temperature

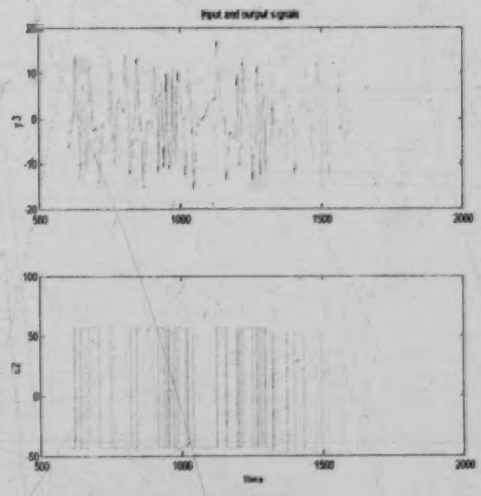

Figure 5.30: After segmentation and detrend of the input and output

doesn't affect the rightness of the reduced process model. Fig. 5.33 and Fig. 5.34 show the fit rate and residual test of g_{22}, while Fig. 5.35 and Fig. 5.36 are for g_{32}. We can conclude from these figures all three process models are good enough.

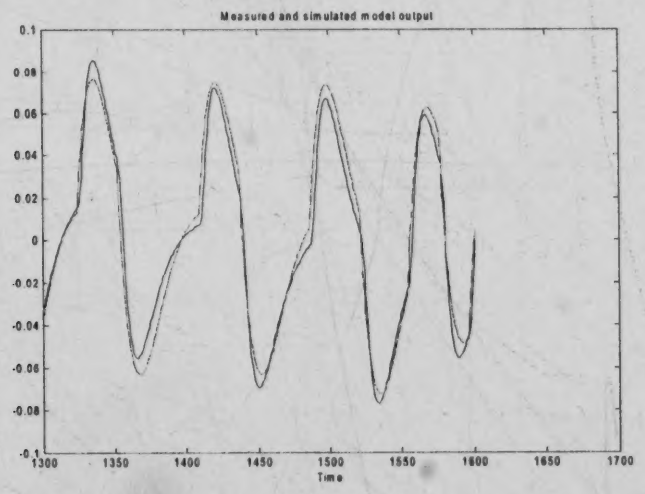

Figure 5.31: The output fit of g_{12}

Step 5 Parameter acquirement and the model transfer from discrete to continuous.

g_{12}

Discrete parameters:

Discrete-time IDPOLY model: $y(t) = [B(q)/F(q)]u(t) + e(t)$

$B(q) = 0.0001114(+-5.579e-006) - 0.0001133(+-5.602e-006)q^-1$

$F(q) = 1 - 1.891(+-0.006027)q^-1 + 0.8967(+-0.006001)q^-2$

Now using the MATLAB function *d2c*, we can transfer the discrete time model to a continuous time model.

Continuous-time IDPOLY model: $y(t) = [B(s)/F(s)]u(t) + e(t)$

$B(s) = 0.0001114s^2 + 0.0001164s - 2.033e-006$

$F(s) = s^2 + 0.1091s + 0.006414$

g_{22}

Discrete parameters

Discrete-time IDPOLY model: $y(t) = [B(q)/F(q)]u(t) + e(t)$

$B(q) = -0.1917(+-0.4002) + 14.8(+-0.8461)q^-1$

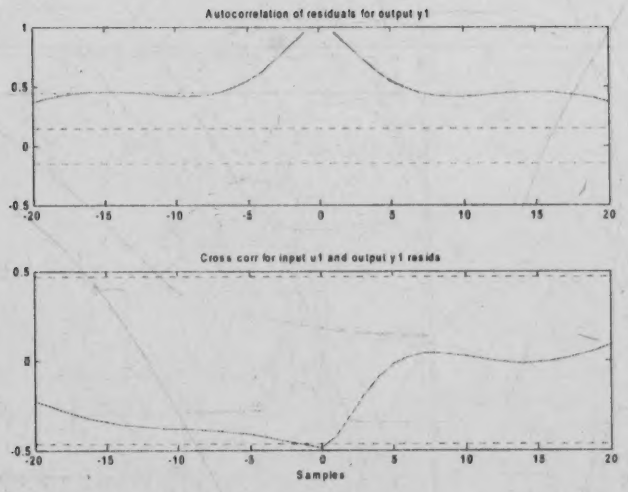

Figure 5.32: The residual test for g_{12}

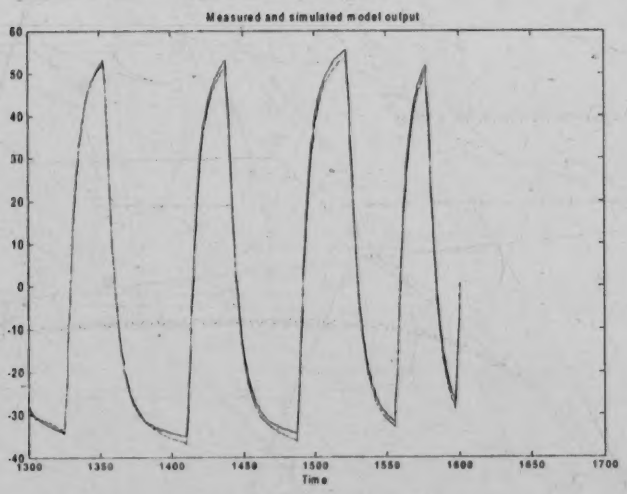

Figure 5.33: The output fit of g_{22}

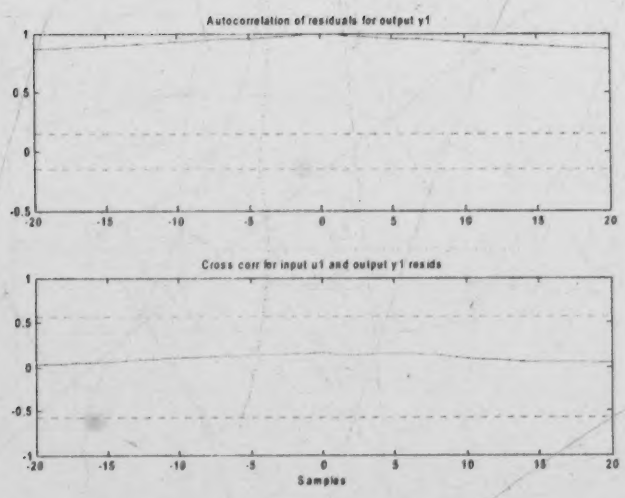

Figure 5.34: The residual test for g_{22}

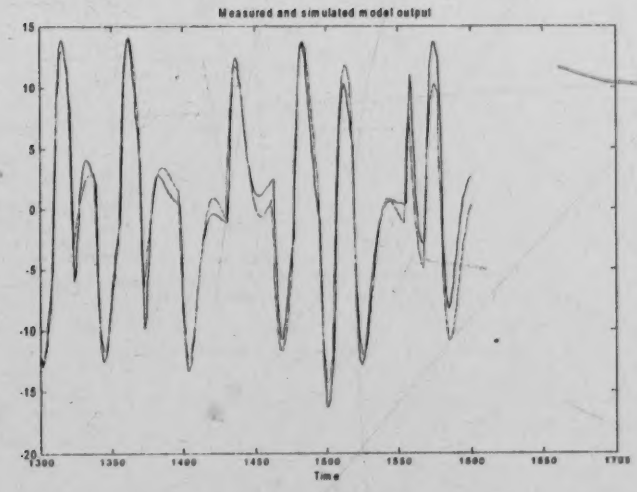

Figure 5.35: The output fit of g_{32}

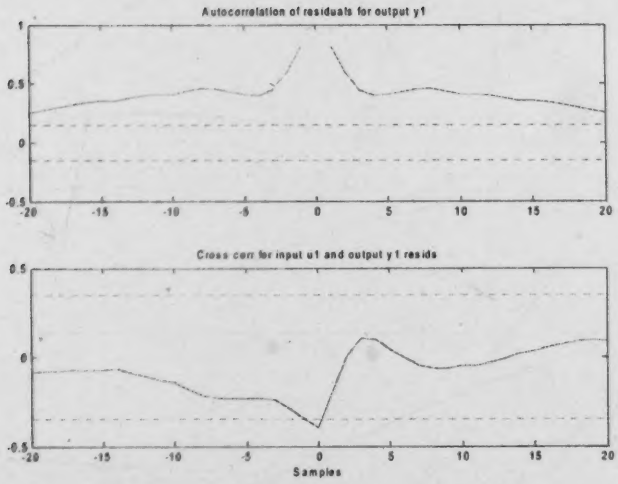

Figure 5.36: The residual test for g_{32}

$$F(q) = \begin{matrix} -27.96(+-0.7441)q^{-}2 + 13.65(+-0.4283)q^{-}3 \\ 1 - 2.177(+-0.01466)q^{-}1 + 1.696(+-0.03385)q^{-}2 \\ -0.5659(+-0.05296)q^{-}3 + 0.1002(+-0.02706)q^{-}4 \end{matrix}$$

Continuous parameter

Continuous-time IDPOLY model: $y(t) = [B(s)/F(s)]u(t) + e(t)$

$B(s) = -0.1917s^4 + 16.17s^3 + 44.25s^2 + 2.912s + 0.9307$

$F(s) = s^4 + 2.3s^3 + 2.427s^2 + 0.5504s + 0.164$

g_{32}

Discrete-time IDPOLY model: $y(t) = [B(q)/F(q)]u(t) + e(t)$

$B(q) = 0.03928(+-0.00105) - 0.0383(+-0.001058)q^{-}1$

$F(q) = 1 - 1.793(+-0.004821)q^{-}1 + 0.8297(+-0.005087)q^{-}2$

Continuous-time IDPOLY model: $y(t) = [B(s)/F(s)]u(t) + e(t)$

$B(s) = 0.03928s^2 + 0.04305s + 0.001079$

$F(s) = s^2 + 0.1866s + 0.04071$

5.2.3 The identification of g_{13}, g_{23} and g_{33}

Step 1 Step response test: u_3–step change, $u_1 = 0$, $u_2 = 0$.

Fig. 5.37 to Fig. 5.39 show all three inputs and Fig. 5.40 to Fig. 5.42 all three outputs.

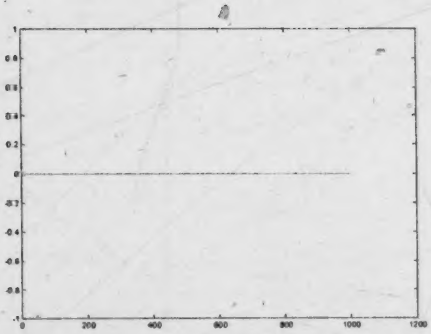

Figure 5.37: Input u_1

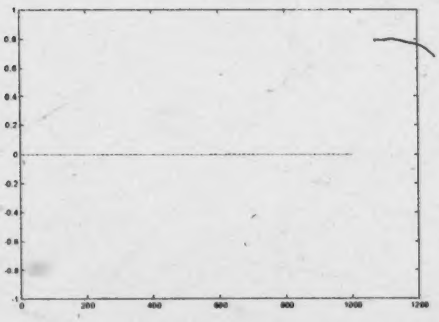

Figure 5.38: Input u_2

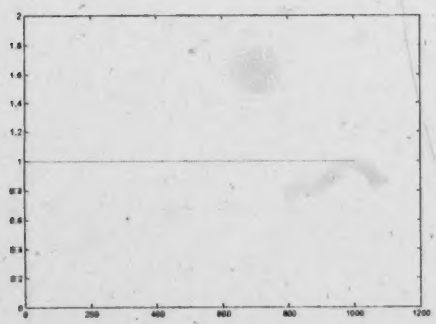

Figure 5.39: Input u_3

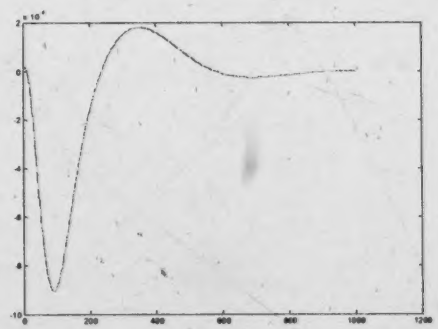

Figure 5.40: Output y_1: Drum level

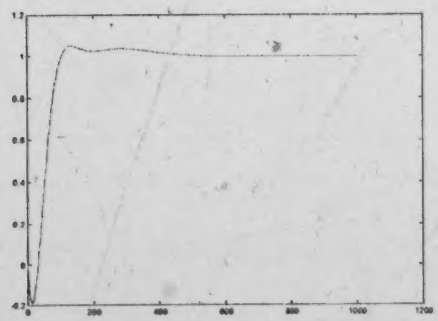

Figure 5.41: Output y_2: Drum pressure

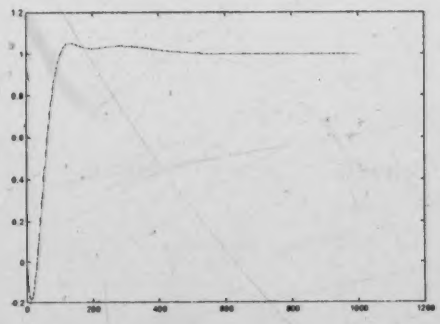

Figure 5.42: Output y_3: Steam temperature

Step 2 RBS input design.

Now according to the figure of y_3, we can calculate its time constant: $\tau = 65(sec)$. So, we take $T_s = 10s$, thus $f = \frac{k*T_s}{\tau\pi} = 0.1$. The following code can be used to produce the RBS input:

$clear\ all;$

$Ts = 10;$

$U = idinput(1000,'rbs',[0\ 0.1],[-50\ 50]);$

$U = [zeros(600,1);U];$

$T = [0 : Ts : Ts * (length(U) - 1)];$

$U = [T'\ U];$

Step 3 Data pre-processing: Segmentation and detrend.

The third reference input signal and the first output signal drum level are shown in Fig. 5.43. The third reference input signal and the second output signal drum pressure are shown in Fig. 5.45. The third input reference signal and the third output signal steam temperature are shown in Fig. 5.47. We use data (601-1300) for identification and data (1301-1600) for validation. The input and output signals after segmentation and detrend are shown in Fig. 5.44 , Fig. 5.46 and Fig. 5.48.

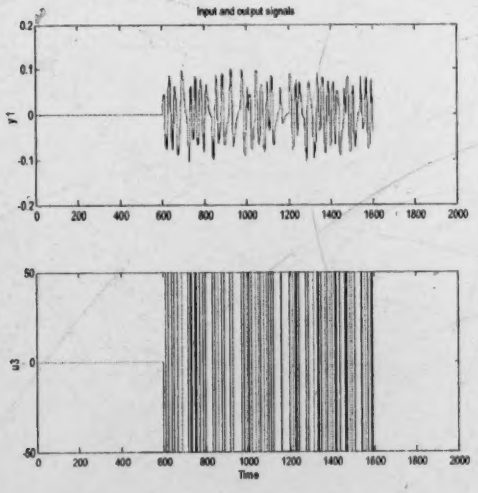

Figure 5.43: The third reference input and drum level output

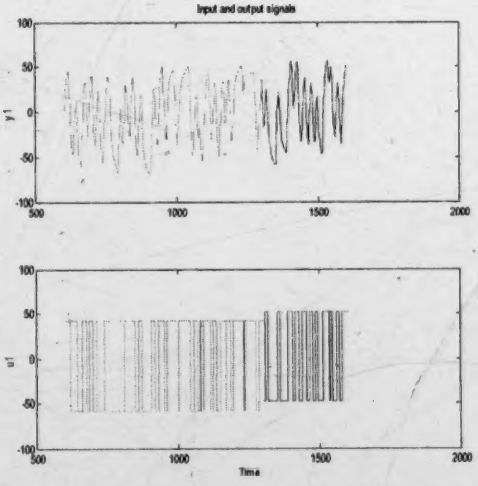

Figure 5.44: After segmentation and detrend of the input and output

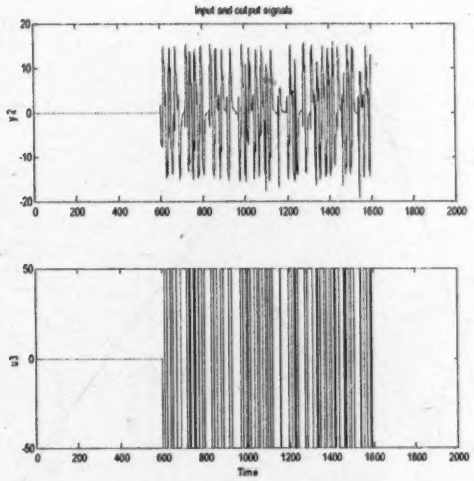

Figure 5.45: The third reference input and drum pressure output

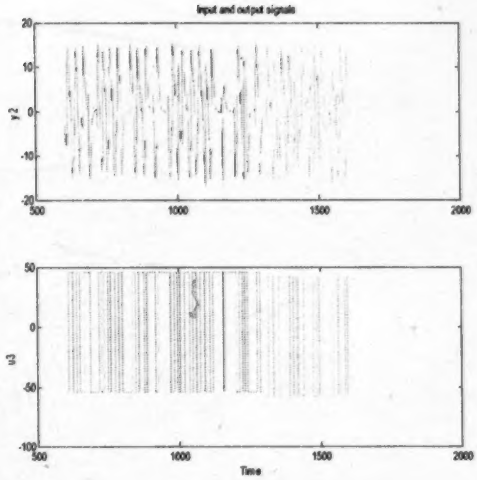

Figure 5.46: After segmentation and detrend of the input and output

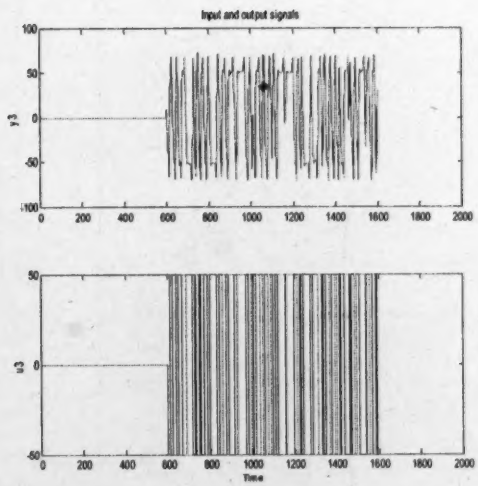

Figure 5.47: The third reference input and steam temperature

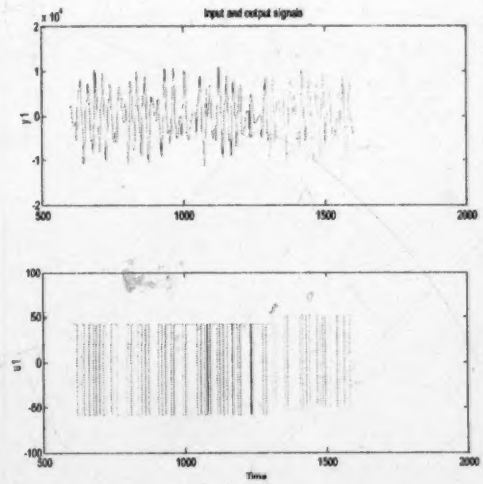

Figure 5.48: After segmentation and detrend of input and output

Step 4 Parameter estimation and model test.

Fig. 5.49 shows the output fit situation of the reduced model and the original of g_{13}. From it, we know the fit rate attains 95.2%. In Fig. 5.50, we check if the reduced model is correct by residual testa. From the bottom part of Fig. 5.50, we know the process model g_{13} is correct holding a 99% confidence interval. The top part of Fig. 5.50 just shows the noise model is not good. This is mainly because of the fact we choose an OE model to identify; it doesn't affect the rightness of the reduced process model. Fig. 5.51 and Fig. 5.52 show the fit rate and residual test for g_{23}, while Fig. 5.53 and Fig. 5.54 are for g_{33}. We can conclude from these figures all three process models are good enough.

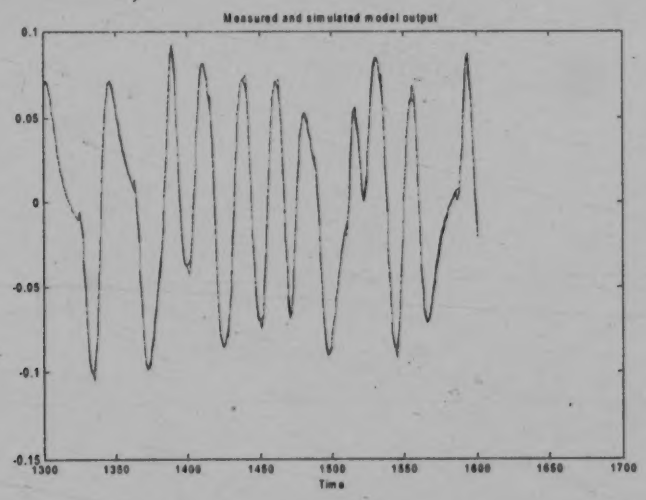

Figure 5.49: The output fit of g_{13}

Step 5 Parameter acquirement and the model transfer from discrete to continuous.

g_{13}

Discrete parameters:

Discrete-time IDPOLY model: $y(t) = [B(q)/F(q)]u(t) + e(t)$

$B(q) = 5.668e-005(+-2.496e-006) - 0.0002356(+-8.206e-006)q^{-}1$
$+0.0002882(+-9.369e-006)q^{-}2 - 0.0001092(+-3.693e-006)q^{-}3$

$F(q) = 1 - 3.497(+-0.00473)q^{-}1 + 4.662(+-0.01214)q^{-}2$
$-2.813(+-0.01072)q^{-}3 + 0.6498(+-0.003266)q^{-}4$

Now using the MATLAB function *d2c*, we can transfer the discrete time model to continuous

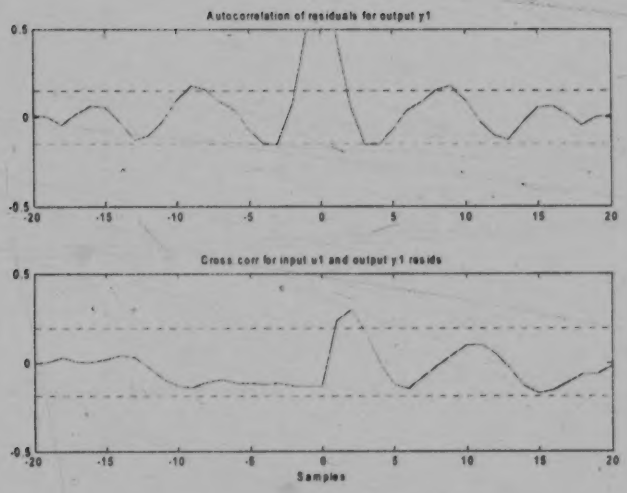

Figure 5.50: The residual test for g_{13}

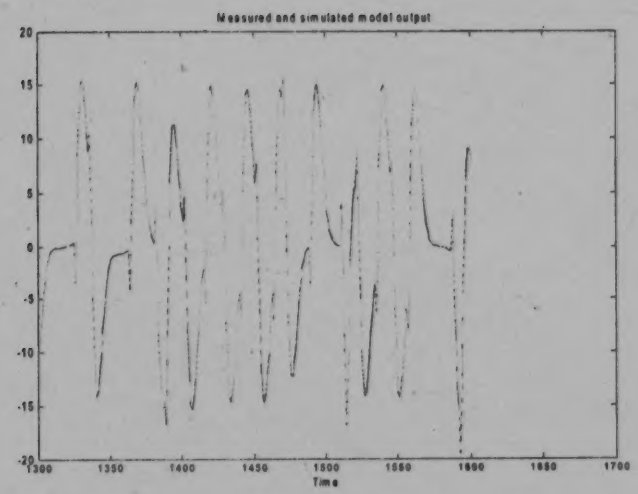

Figure 5.51: The output fit of g_{23}

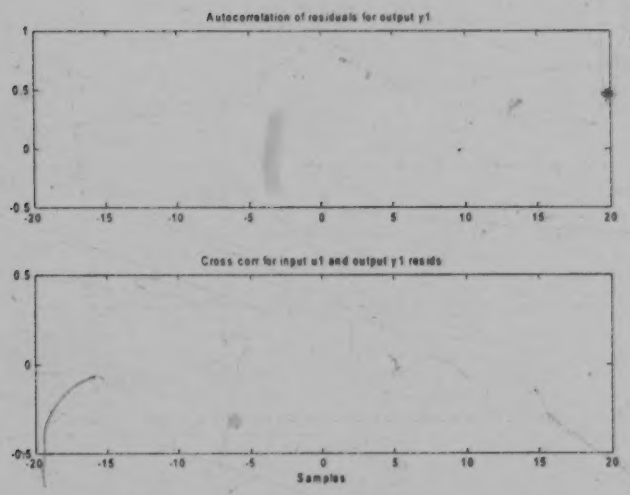

Figure 5.52: The residual test for g_{23}

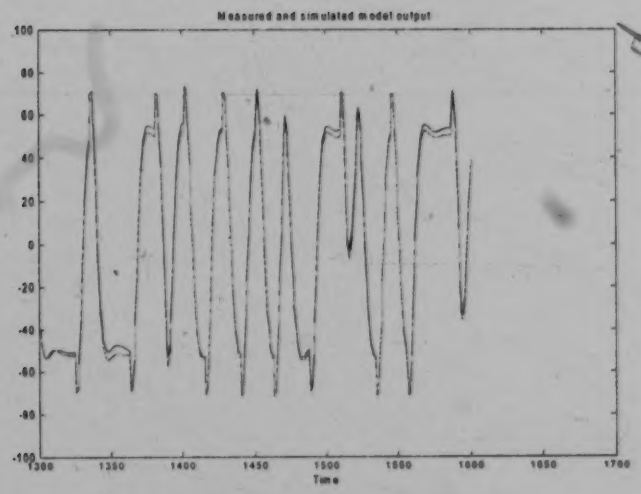

Figure 5.53: The output fit of g_{33}

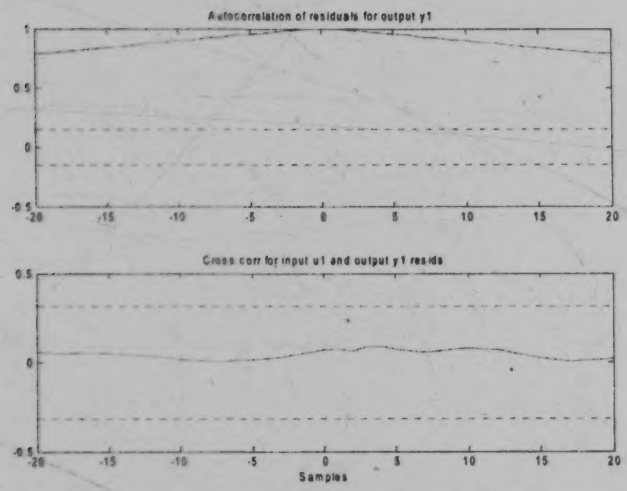

Figure 5.54: The residual test for g_{33}

time model.

Continuous-time IDPOLY model: $y(t) = [B(s)/F(s)]u(t) + e(t)$

$B(s) = 5.668e-005s^4 + 3.042e-005s^3 - 8.192e-005s^2 - 1.637e-005s + 5.768e-008$

$F(s) = s^4 + 0.4311s^3 + 0.1797s^2 + 0.0197s + 0.001908$

g_{23}

Discrete pa meters

Discrete-time IDPOLY model: $y(t) = [B(q)/F(q)]u(t) + e(t)$

$B(q) = 0.0001408(+-0.0004395) - 0.03826(+-0.0008709)q^-1$
$+0.1331(+-0.000672)q^-2 - 0.09515(+-0.0002761)q^-3$

$F(q) = 1 - 1.832(+-0.00341)q^-1 + 0.9655(+-0.009748)q^-2$
$+0.004776(+-0.01097)q^-3 - 0.09099(+-0.004636)q^-4$

Continuous parameter

Continuous-time IDPOLY model: $y(t) = [B(s)/F(s)]u(t) + e(t)$

$B(s) = 0.0001408s^4 - 0.09569s^3 - 0.03035s^2 + 0.105s - 0.0002529$

$F(s) = s^4 + 2.397s^3 + 1.83s^2 + 0.6601s + 0.0863$

g_{31}

Discrete-time IDPOLY model: $y(t) = [B(q)/F(q)]u(t) + e(t)$

$B(q) = 0.1367(+-0.6574) + 26.99(+-1.898)q^-1$

$$-57.49(+-2.029)q^-2+30.44(+-0.8022)q^-3$$
$$F(q)=1-2.766(+-0.01618)q^-1+2.797(+-0.04849)q^-2$$
$$-1.213(+-0.04928)q^-3+0.1891(+-0.01715)q^-4$$

Continuous-time IDPOLY model: $y(t)=[B(s)/F(s)]u(t)+e(t)$

$B(s)=0.1367s^4+24.91s^3+61.42s^2-6.874s+0.1811$

$F(s)=s^4+1.665s^3+0.8306s^2+0.1486s+0.01757$

We use Table 5.1 to summarize fit rates and order situations of all nine elements.

Table 5.1: Fit rate and order reduction situation

	g_{11}	g_{12}	g_{13}	g_{21}	g_{22}	g_{23}	g_{31}	g_{32}	g_{33}
The original model order	19	19	19	19	19	19	19	19	19
The reduced model order	4	2	4	4	4	4	4	2	4
Fit rate(%)	97.6	78.0	95.2	85.0	95.6	95.6	90.5	73.5	94.7

5.3 MPC design and simulation

For the compensated and reduced plant, we set the following cost function:

$$J=\sum_{i=1}^{N_2}\|w(k+i)-\hat{y}(k+i)\|^2_{Q(i)}+\sum_{i=1}^{N_u}\|\Delta u(k+i-1)\|^2_{R(i)}, \tag{5.14}$$

subject to following constraints

$$u_{min}\le u(k)\le u_{max}, \tag{5.15}$$

where w is the reference output, $\hat{y}$ the predicted output, N_2 the output prediction horizon, N_u the control horizon, Q, R are the weighting factors, and Δu is the input change from the previous step. To solve this constrained problem, we need to translate input amplitude constraints into ones on Δu. It is easy to convert constraints on inputs in our plant into the following form:

$$F\begin{bmatrix}U(k)\\1\end{bmatrix}\le 0, \tag{5.16}$$

where $U(k)=[\hat{u}(k/k)^T,...,\hat{u}(k+N_u-1/k)^T]^T$. Let F has the form

$$F=[F_1,F_2,...,F_{N_u},f],$$

where each F_i has size $q*m$, and f has $q*1$. Thus (5.16) can be written as

$$\sum_{i=1}^{N_u}F_i\hat{u}(k+i-1/k)+f\le 0. \tag{5.17}$$

Because the control horizon is N_u, we have

$$\hat{u}(k+i-1/k) = u(k-1) + \sum_{j-0}^{i-1} \Delta\hat{u}(k+j/k). \tag{5.18}$$

We substitute Eq. (5.18) into Eq. (5.17),

$$\sum_{j-1}^{N_u} F_j \Delta\hat{u}(k/k) + \sum_{j-2}^{N_u} (F_j \Delta\hat{u}(k+1/k) + ... + F_{N_u}\Delta\hat{u}(k+N_u-1/k)) + \sum_{j=1}^{N_u} F_j u(k-1) + f \leq 0. \tag{5.19}$$

Now define $\Gamma_i = \sum_{j=i}^{N_u} F_j$ and $\Gamma = [\Gamma_1, ..., \Gamma_{N_u}]$. Then Eq. (5.19) can be written as

$$\Gamma\Delta U(k) \leq -\Gamma_1 u(k-1) - f. \tag{5.20}$$

Now the problem becomes to minimize the cost function J subject to the inequality constraint in (5.20). That is a standard optimization problem known as the quadratic programming (QP) problem, and standard MATLAB function is available, e.g., MATLAB functions.

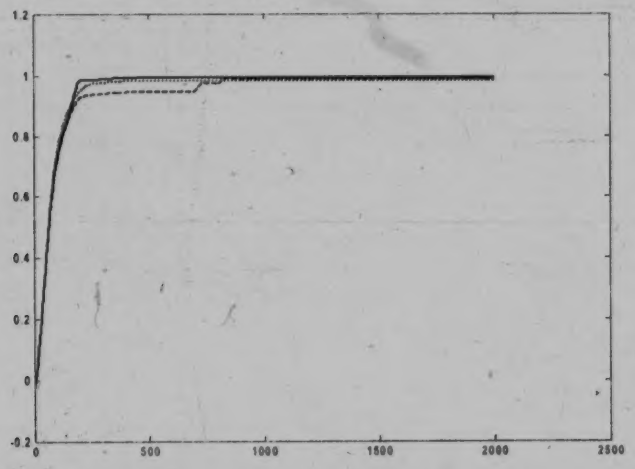

Figure 5.55: Step response of the drum level with the linear model

Similar with the last chapter, two different cost functions are used to design controllers to show the effect of LEC. The first one is given by

$$J_1 = \sum_{j=1}^{N_2} q_1(\hat{y}_1 - w_1)^2 + \sum_{j=1}^{N_2} q_2(\hat{y}_2 - w_2)^2 + \sum_{j=1}^{N_2} q_3(\hat{y}_3 - w_3)^2 + \sum_{j=1}^{N_u} r(\Delta U)^2,$$

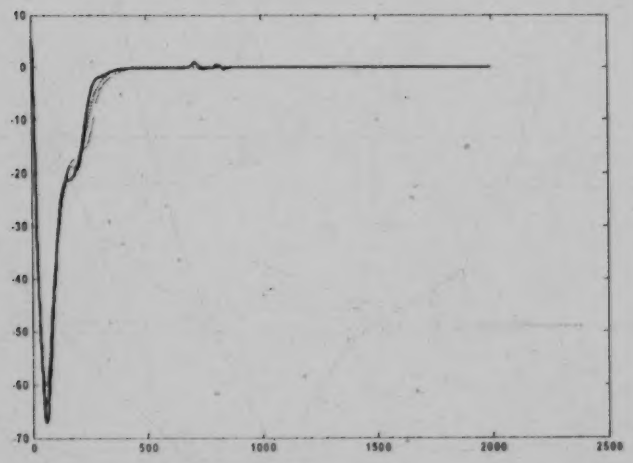

Figure 5.56: Step response of the drum pressure with the linear model

where $\hat{y}_1, \hat{y}_2$ and $\hat{y}_3$ are the predicted outputs (thus J_1 does not account for the damage variables); the second is

$$J_2 = \sum_{j=1}^{N_2} q_1(\hat{y}_1 - w_1)^2 + \sum_{j=1}^{N_2} q_2(\hat{y}_2 - w_2)^2 + \sum_{j=1}^{N_2} q_3(\hat{y}_3 - w_3)^2 + \sum_{j=1}^{N_2} q_4(\sigma)^2 + \sum_{j=1}^{N_u} r(\Delta U)^2$$

which includes the fatigue variable the stress (σ), the output of damage model I. Because the damage dynamics is nonlinear, linearization should be performed first at the normal load conditions. Controllers 1 and 2 are designed by minimizing J_1 and J_2, respectively, using MPC theory, setting q_4 to be 0.3 and other weightings unity, $N_2 = 6$ and $N_u = 4$. Because of the way cost functions are defined, controller 2 is an LEC; while controller 1 is not.

Simulation results with linear model are shown in Fig. 5.55 to Fig. 5.58, and with nonlinear model are in Fig. 5.59 to Fig. 5.62. In all figures, dotted lines represent results of the original system, solid lines results when the controller 1 is applied and dashed lines results when the controller 2 is applied. From simulation results, both the controller 1 and 2 improve the performance of system and reduce the accumulated damage, while controller 2 (LEC) reduces the accumulated damage more than controller 1 does.

To quantify the loss in performance, now we define

$J_1^{\dagger} \triangleq$ the contribution of the three outputs by the optimal controller by minimizing J_1

$$= \int (q_1 y_1^2 + q_2 y_2^2 + q_3 y_3^2) dt,$$

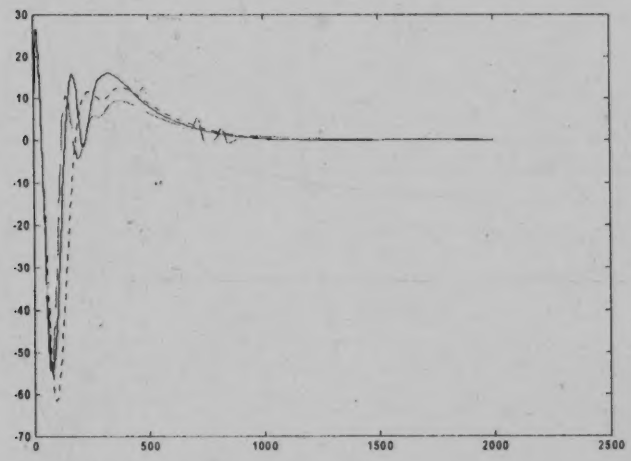

Figure 5.57: Step response of the steam temperature with the linear model

and

$J_2^\dagger \triangleq$ the contribution of the three outputs by the optimal controller by minimizing J_2

$$= \int (q_1 y_1^2 + q_2 y_2^2 + q_3 y_3^2) dt,$$

where So, the relative loss L_r in performance can be calculated as

$$L_r = \frac{J_2^\dagger - J_1^\dagger}{J_1^\dagger}. \tag{5.21}$$

By Eq. 5.21, for the linear model we have

$$L_r = \frac{2.7176 * 10^5 - 2.4861 * 10^5}{2.4861 * 10^5} = 9.31\%,$$

and for the nonlinear model,

$$L_r = \frac{3.145 * 10^5 - 2.927 * 10^5}{2.927 * 10^5} = 7.44\%.$$

With these losses in performance, we gain about 50% reduction in accumulated damage both for the linear and the nonlinear models.

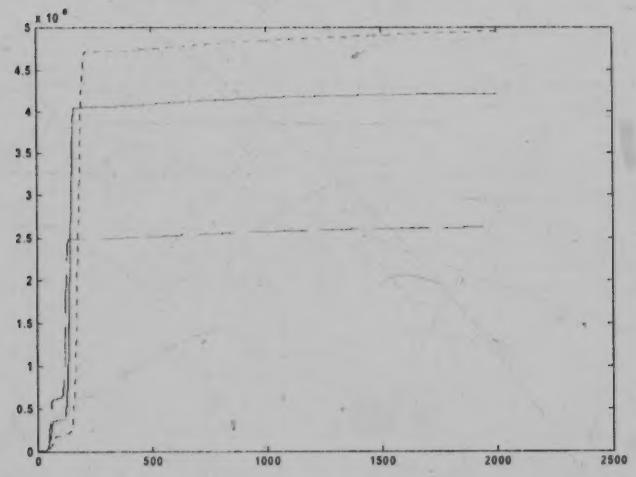

Figure 5.58: Comparison of the accumulated damage with the linear model

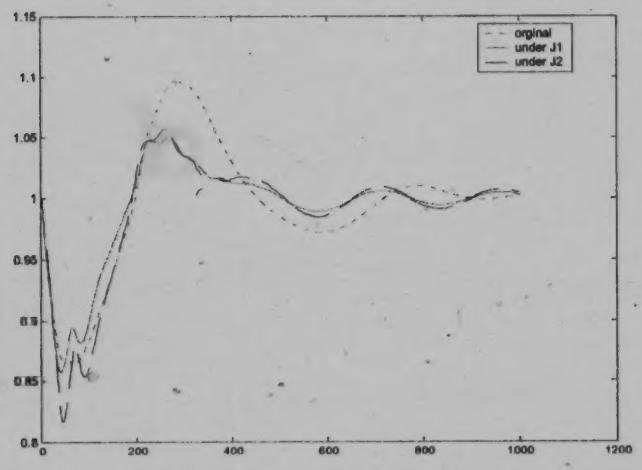

Figure 5.59: Step response of the drum level with the nonlinear model

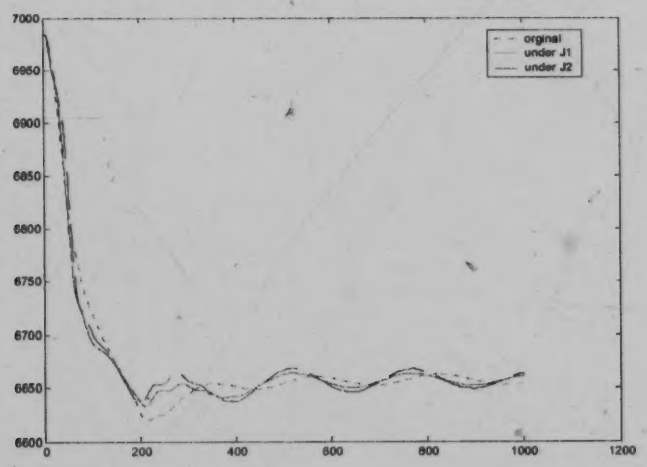

Figure 5.60: Step response of the drum pressure with the nonlinear model

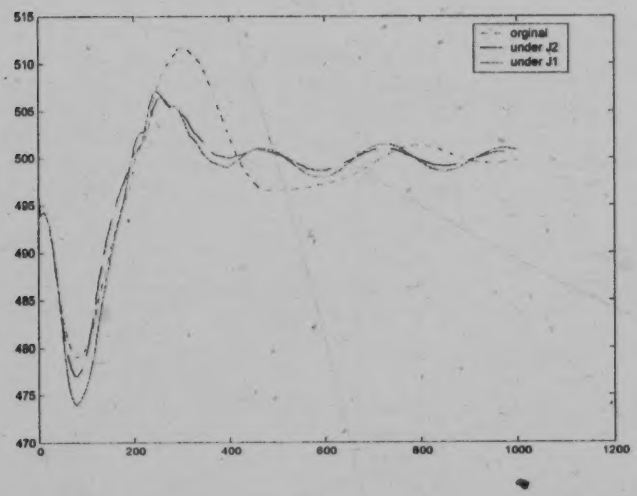

Figure 5.61: Step response of the steam temperature with the nonlinear model

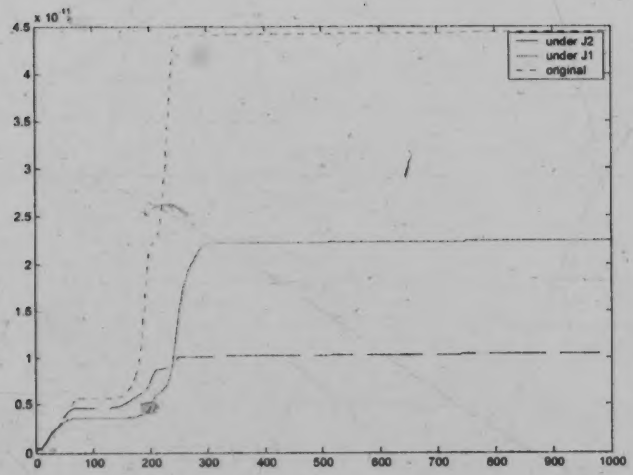

Figure 5.62: Comparison of the accumulated damage with the nonlinear model

Chapter 6

Conclusions and Future work

LEC is a new branch of engineering which integrates system sciences and mechanics of materials. The challenges in this area are mainly from the two facts: first, the knowledges of both control systems and materials are needed; second, it emphasizes on practical applications, however, various physical limits from industrial environments make it very difficult to obtain some required data.

In this thesis both important topics in LEC, fatigue damage modeling and life extending controller design, are discussed. The main contributions are as follows:

- We proposed a hierarchical structure for the implementation of our LEC scheme, which allows existing control systems to remain in operation, making our plan more acceptable for industrial application already in operation. Meanwhile this structure includes two damage models of the plant, damage model I (for feedback control) and damage model II (for life prediction).

- Neither P-M nor nonlinear fatigue modeling techniques are suitable for on-line control implementation. Although the continuous fatigue damage model proposed by Ray is one step closer to control implementation, it is still difficult to use in on-line control. We incorporated an improved Rainflow cycle counting method with Ray's modeling theory to build a damage model for on-line model-based life extending control.

- We formulated our specific research problem according to the linear quadratic regulator theory, and applied LQR theory to design an LEC for a boiler-turbine system in a co-generation system.

- We modified and formulated our life extending control design problem for industrial boiler systems using model-based predictive control theory and applied MPC control

theory to design an LEC for a boiler-turbine system in the Syncrude co-generation system.

- We simulated our LEC closed-loop systems using linear and nonlinear models, and demonstrated the advantages of using LEC over design without LEC.

The future work includes:

- We have constructed two kinds of fatigue damage models. Now the problem is: Are these models good? To validate our models against real data is an important and critical step. However we also should notice that it is very difficult to acquire the real data which are related to the life of a boiler-turbine system. This is the reason why it is difficult to evaluate an LEC design. Zhang and Ray (1998) [55] validated their damage model with a laboratory test apparatus. That should has some implication in our future research.

- The damage models we have constructed are in the transfer function form, to design an LEC with LQR, we have to transfer our models into the state space form. So building a state space fatigue damage model directly is a better choice.

- From the process of our damage modeling, the model necessarily contains errors, therefore the robust control design for LEC is expected to produce better results.

Bibliography

[1] American Society for Testing and Materials, Annual Book of ASTM Standards, Section 3: Metals test methods and analytical procedures, Vol. 03.01-Metals-Mechanical Testing; Elevated and Low-Temperature Tests, ASTM, 1986, pp. 836-848.

[2] H. Anzai and T. Endo, On-site indication of fatigue damage under complex loading, Int. J. Fatigue, No. 1, 1979, pp. 49-57.

[3] J. A. Bannantine, J. J. Comer and J. L. Handrock, Fundamentals of Metal Fatigue Analysis, Prentice-Hall, 1990.

[4] J. A. Bemporad, Reducing conservativeness in predictive control of constrained systems with disturbances, Proceedings of the 37th IEEE Conference on Decision & Control, Dec. 1998, pp. 1385-1389.

[5] E. F. Camacho and C. Bordons, Model Predictive Control, Springer, 1999.

[6] P. Campo and M. Morari, Robust model predictive control, Proceedings of the American Control Conference, June 1987, pp. 1021-1026.

[7] T. Chen and B. A. Francis, Optimal Sampled-Data Control Systems, Springer, 1995.

[8] D. W. Clarke, C. Mohtadi and P. S. Tuffs, Generalized predictive control: Part II–extensions and interpretations, Automatica, 23(2), 1987, pp. 149-160.

[9] C. R. Culter and B. L. Ramaker, Dynamic matrix control–a computer control algorithm, Proceedings of Joint Automatic Control Conference, 1980, WB5-B.

[10] X. Dai and A. Ray, Damage-mitigating control of a reusable rocket engine: Parts I and II, ASME Journal of Dynamic Systems, Measurement and Control, Vol. 118, No. 3, 1996, pp. 401-415.

[11] N. E. Dowling, A discussion of methods for estimating fatigue life, Proceedings of the SAE Fatigue Conference, April 1982, pp. 161-175.

[12] N. E. Dowling, Fatigue life prediction for complex load versus time histories, ASME Journal of Engineering Materials and Technology, Vol. 105, 1983, pp. 206-214.

[13] S. D. Downing and D. F. Socie, Simple rainflow counting algorithms, Int. J. Fatigue, January 1982, pp. 31-38.

[14] F. Ellyin, Effect of tensile-mean-strain on plastic strain energy and cyclic response, ASME Journal of Engineering Materials and Technology, Vol. 107, 1985, pp. 119-125.

[15] F. Ellyin, Fatigue Damage, Crack Growth and Life Prediction, Chapman & Hall, 1997.

[16] A. M. Freudenthal and T. J. Dolan, The Character of the Fatigue of Metal, Fourth Progress Report, February 1948.

[17] C. E. Garcia and M. Morari, Internal model control-1. A unifying review and some new results, Industrial & Engineering Chemistry Process Research and Development, 1982, pp. 308-323.

[18] C. E. Garcia, D. M. Preit and M. Morari, Model predictive control: Theory and practice-a survey, Automatica, Vol. 25, No. 3, 1989, pp. 335-348.

[19] H. J. Grover, S. A. Gordon and L. R. Jackson, Fatigue of Metals and Structures, US. Government Printing Office, 1954.

[20] M. S. Holmes, Damage Mitigating Control of Mechanical Systems, A Thesis of the Pennsylvania State University, August 1997.

[21] M. S. Holmes and A. Ray, Fuzzy damage mitigating control of mechanical structures, ASME Journal of Dynamic Systems, Measurement and Control, Vol. 120, 1998, pp. 249-256.

[22] B. Huang and S.L. Shah, Performance Assessment of Control Loops: Theory and Applications, Springer Verlag, 1999.

[23] P.T. Kallappa, M. Holmes and A. Ray, Life extending control of fossil power plants for structural durability and performance enhancement, Automatica, Vol. 33, No. 6, 1997, 1101-1118.

[24] L. Ljung, System Identification: Theory for the User, Prentice-Hall, 1999.

[25] C. F. Lorenzo, Continuum Fatigue Damage Modeling for Use in Life Extending Control, NASA Technical Memorandum 106691, August 1994.

[26] S. Lu and B. Wilson, On-line stress calculation and life monitoring systems for boiler components, Trans. Inst. MC, Vol. 20, No. 1, 1998, pp. 20-28.

[27] E. S. Machlin, Dislocation Theory of Fatigue of Metals, NACA Report 929, 1949.

[28] J. M. Maciejowski, Predictive Control with Constraints, Prentice-Hall, 2002.

[29] H. J. Marquez, Nonlinear Control Systems: Analysis and Design, Wiley-Interscience, 2003.

[30] R. K. Mehra, R. Rouhani, J. Eterno, J. Richalet and A. Rault, Model algorithmic control: Review and recent development, Engng Foundation Conference on Chemical Process Control II, 1982, pp. 287-310.

[31] M. A. Miner, Cumulative damage in fatigue, Journal of Applied Mechanics, Vol. 12, 1945, pp. A159-A164.

[32] T. E. Noll, S. Donley, G. Graham, T. Harris, I. Kaynes, B. Lee and J. Sparrow, Impact of active controls technology on structural integrity, 32nd AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference, April 1991, pp. 1869-1878.

[33] E. Orowan, Theory of the fatigue of metals, Proceedings of London Royal Society, Vol. 171A, 1939 pp. 79-105.

[34] R. Patankar and A. Ray, Damage mitigating controller design for structural durability, IEEE Transactions on Control Systems Technology, Vol. 7, No. 5, Sep. 1999, pp. 606-612.

[35] G. Poncia and S. Bittanti, Multivariable model predictive control of a thermal power plant with built-in classical regulation, INT. J. Control, Vol. 74, No. 11, 2001, pp. 1118-1130.

[36] L. P. Pook, The Role of Crack Growth in Metal Fatigue, J. W. Arrowsmith Ltd, 1983.

[37] A. Ray, X. Dai, M. Carpino and C.F. Lorenzo, Damage-mitigating control - An interdisciplinary thrust between controls and material science. Proceedings of the American Control Conference, June 1994, pp. 3449-3453.

[38] A. Ray, M. Wu, M. Carpino and C. F. Lorenzo, Damage-mitigating control of mechanical systems: Part I–conceptual development and model formulation, ASME Journal of Dynamic Systems, Measurement, Control, Vol. 116, 1994, pp. 437-447.

[39] A. Ray, M. Wu, M. Carpino and C. F. Lorenzo, Damage-mitigating control of mechanical systems: Part II–formulation of an optimal policy and simulation, ASME Journal of Dynamic Systems, Measurement, Control, Vol. 116, 1994, pp. 448-455.

[40] A. Ray and S. Tangirala, Stochastic modeling of fatigue crack dynamics for on-line damage estimation and life prediction, IEEE Trans. on Control Systems Technology, Vol. 4, No. 4, 1996, pp. 443-451.

[41] J. Richalet, A. Rault, J. L. Testud and J. Papon, Model predictive heuristic control: Application to industrial processes, Automatica, 14(2), 1978, pp. 413-428.

[42] R. D. Rink, A. White and R. Leung, SYNSIM: A Computer Simulation Model for the Mildred Lake Steam/Electrical System of Syncrude Canada Ltd., Technical Report, University of Alberta, May, 1996.

[43] G. Schucktanz and M. Miksch, On-line fatigue monitoring of reactor components, Proceedings of International Conference on Life Assessment and Extension, 1988, Paper 3.5.2.

[44] S. L. Shah, C. Mohtadi and D. W. Clarke, Advances in model-based predictive control, Proceedings IEE, Part D, 140(4), 1993.

[45] D. F. Socie, Fatigue life prediction using local stress-strain concepts, Experimental Mechanics, 17:2, Feb. 1977, pp. 50-56.

[46] T. Soderstrom and P. Stoica, System Identification, Prentice-Hall, 1989

[47] N. R. Sripada and D. G. Fisher, Multivariable optimal constrained control algorithm: Part 1– formulation and application, Proceedings of International Conference on Modeling and Control, 1985.

[48] S. Tangirala, Stochastic Modeling of Fatigue Damage for On-Line Monitoring and Life Extending Control, Ph.D Thesis in the Pennsylvania State University, May, 1996.

[49] S. Tangirala, M. Holmes and A. Ray, Life-extending control of mechanical structures: A feedback approach, Automatica, Vol. 34, Jan. 1998, pp. 3-14.

[50] W. Tan, H. J. Marquez, T. Chen and R. K. Gooden, H_∞ control design for an industrial boiler, Proceedings of the American Control Conference, June 2001, pp. 2537-2542.

[51] D. M. White, Robust Controller Systems Utilizing *Khantonov's* Theorem, Master Thesis in UA, 1998.

[52] L. B. William, Mondern Control Theory, Prentice-Hall, 1991.

[53] A. B. Wilson, Stress monitoring systems for safeguarding boiler components during rapid temperature transients, Proceedings of IMechE Conference on The Two-Shifting of CCGT and Fossil Fired Steam Plant, 1995, pp. 43-48.

[54] B. E. Ydstie, Extended horizon adaptive control, A Bridge Between Control Science And Technology, Pergamon Press, 1984, pp. 134-137.

[55] H. Zhang and A. Ray, Robust damage-mitigating control of mechanical structures: Experimental verification on a test apparatus, ASME Journal of Dynamic Systems, Measurement, and Control, Vol. 121, No. 3, 1999, pp. 377-385.

[56] K. Zhou, J. Doyle and K. Glover, Robust and Optimal control, Prentice-Hall, 1996.